No. 2014-1002

United States Court of Appeals for the Federal Circuit

IN RE: APPLE INC.

Appeal from the United States Patent and Trademark Office,
Patent Trial and Appeal Board,
Reexamination No. 90/011,287.

BRIEF FOR APPELLANT APPLE INC.

PETER J. YIM
MORRISON & FOERSTER LLP
425 Market Street
San Francisco, CA 94105
Telephone: 415.268.7000

DEANNE E. MAYNARD
MARC A. HEARRON
MORRISON & FOERSTER LLP
2000 Pennsylvania Avenue, N.W.
Washington, D.C. 20006
Telephone: 202.887.8740

Counsel for Appellant Apple Inc.

January 31, 2014

CERTIFICATE OF INTEREST

Counsel for the appellant Apple Inc. certifies the following:

1. The full name of every party or amicus represented by me is:

Apple Inc.

2. The name of the real party in interest (if the party named in the caption is not the real party in interest) represented by me is:

N/A

3. All parent corporations and any publicly held companies that own 10% or more of the stock of the party or amicus curiae represented by me are:

N/A

4. The names of all law firms and the partners or associates that appeared for the party or amicus now represented by me in the trial court or are expected to appear in this court are:

Morrison & Foerster LLP: Marc A. Hearron, Brian B. Ho, Deanne E. Maynard, Peter J. Yim

Novak Druce Connolly Bove + Quigg LLP: Vincent M. DeLuca, Jurg L. Domenig, Tracy W. Druce, Thomas M. Isaacson, Stephen Marcus, Benjamin B. Matthew, Brian K. McKnight, Shebli Mikailli, James P. Murphy, Michael W. O'Neill, Thomas P. Pavelko, Henry A. Petri, Donald J. Quigg, Eduardo J. Quinones, Juan I. Rossi, Joseph R. Weatherbee

Dated: January 31, 2014

/s/ Deanne E. Maynard
Deanne E. Maynard

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Page(s)

CASES

OTHER AUTHORITIES

STATEMENT OF RELATED CASES

This action has not previously been before this or any other appellate court. Counsel for appellant Apple Inc. are aware of the following pending cases that may be affected directly by this Court's decision, all of which involve the same patent but not the claims at issue here:

- *Apple Inc. v. Motorola, Inc.*, No. 11-cv-08540 (N.D. Ill.), *appeal docketed*, Nos. 2012-1548, -1549 (Fed. Cir.) (Rader, C.J., Prost, Reyna, J.J.) (oral argument heard Sept. 11, 2013),
- *Motorola Mobility Inc. v. Apple Inc.*, No. 10-cv-00867 (D. Del.), *transferred to* No. 12-cv-00079 (N.D. Ill.), and
- *Apple Inc. v. Samsung Electronics Co., Ltd.*, No. 5:12-cv-00630-LHK (N.D. Cal.), *preliminary injunction rev'd*, No. 12-1507 (Fed. Cir. Oct. 11, 2012) (Prost, Moore, Reyna, J.J.).

Counsel are unaware of any other pending case that will affect or be affected directly by this Court's decision.

JURISDICTIONAL STATEMENT

The Patent Trial and Appeal Board ("Board") of the United States Patent and Trademark Office ("PTO") had jurisdiction over the appeal from the examiner's final reexamination ruling under 35 U.S.C. §§ 6(b)(2) and 134(b). The Board mailed its decision on June 20, 2013. Appellant Apple Inc. ("Apple") appealed on August 20, 2013. This Court has jurisdiction under 28 U.S.C. § 1295(a)(4)(A).

STATEMENT OF THE ISSUES

1. Whether the Board erroneously rejected claims of U.S. Patent No. 5,946,647 ("'647 patent"), as well as newly added claims 25-31, in light of the Pensoft reference.

2. Whether the Board erroneously rejected claims of the '647 patent in light of the Nokia reference.

INTRODUCTION

In rejecting the claims at issue here, the PTO ignored the novelty of the invention claimed in Apple's '647 patent. Apple's claims are directed to specific advancements over the prior, conventional technology. Yet the PTO reexamined the '647 patent and concluded that two examples of the old conventional technology anticipated certain claims of the patent. Neither of those prior art references, however, disclosed the particular advancements that the '647 patent contributed to the field. Instead of comparing the references to the specifics of the

claims, the PTO overly simplified the invention into a collection of disparate elements—data detection, linking, and actions. Yet the claims require that these elements, and more, be arranged in a particular manner not found in either reference.

As the patent explains, it was known in the prior art that computer programmers could create software routines called "parsers" that could search out particular items of information in computer data. For example, a parser could scan through an e-mail to identify dates, addresses, telephone numbers, e-mail addresses, and other such items in the e-mail. Those conventional systems, however, did not create linkages between the item and actions that could be automatically performed on the item.

The invention in the '647 patent overcame those limitations. In the '647 patent, when the computer detects an item of information (referred to in the patent as a "structure"), the computer will create a link or association between the structure and one or more actions (i.e., computer subroutines) that can be performed on the structure. The computer enables the user to select a detected structure, and also to choose an action from the one or more actions that were linked to the detected structure. The computer then automatically executes that action.

Even if the two prior art references relied on by the PTO could detect structures, the two references do not disclose what Apple added—linking an action to the structure, enabling selection of a linked action, and executing the linked action selected by the user.

One of the prior art references, referred to as "Pensoft," is a user manual that simply discloses a conventional relational database that links together items of information such as people, companies, appointments, and notes. A "link" in Pensoft is defined as a relationship between two items of information. For example, Pensoft establishes a link between a person's name and the company for whom the person works. But unlike the invention of the '647 patent, Pensoft does not disclose linking an *action* to a detected item of information.

The second reference, referred to as "Nokia," discloses an early mobile telephone capable of receiving and displaying text messages and searching for potential telephone numbers in a received message. But Nokia does not disclose the linking of an action to the telephone number. Indeed, the concept of linking is nowhere to be found in the Nokia reference. The telephone disclosed in Nokia is merely an example of the conventional systems criticized in the '647 patent as being capable of detecting items of information but not capable of creating the linkages that allow for automatic execution of actions on the detected information.

Neither Pensoft nor Nokia discloses each and every limitation of the reexamined claims. As such, neither reference anticipates. The PTO's rejection of claims 13 and 15-31 of the '647 patent thus should be reversed.

STATEMENT OF THE CASE

A. Apple's '647 patent

The claims of Apple's '647 patent at issue here are directed to methods for performing actions on structures in computer-generated data. A58(col.1:9-11).

The "Background of the Invention" section of the specification discloses that a significant amount of computer data "contains recognizable structures," such as "phone numbers, e-mail addresses, post-office addresses, zip codes and dates." A58(col.1:13-16). In the art that existed before the '647 patent, it was possible for computer programmers to create "pattern analysis units, such as parsers," to automatically identify these "structures" in computer data. A58(col.1:25-27).

As used in the patent, the term "pattern" refers to data used "to recognize information in a document, such as dates, addresses, phone numbers, names, etc." A58(col.1:27-31). "The term 'structure' refers to an instantiation of a pattern in the document." A58(col.1:31-32). For example, "a 'date' pattern will recognize the structure 'Oct. 31, 1995.'" A58(col.1:33-34).

Conventional computer programs merely could "identify structures in computer data." A58(col.1:36-37). They did "not enable automatic performance

of an *action* on an identified structure." A58(col.1:37-38) (emphasis added). For example, conventional computer programs could "search for particular structures, such as telephone numbers," from "a long e-mail message." A58(col.1:38-41). They did not, however, enable automatic performance of "an action on the structure, such as moving the number to an electronic telephone book." A58(col.1:42-43). Instead, performing that action required manually "cutting the structure from the e-mail message, locating and opening the electronic telephone book application program, pasting the structure into the appropriate field, and closing the application program." A58(col.1:44-47). According to the patent, "despite the fact that computer systems are getting faster and more efficient, this procedure is still tedious and cognitively disruptive." A58(col.1:47-50).

The invention disclosed in the '647 patent overcame those limitations. In general, the patent is directed to methods comprising five steps (and computer-based systems that perform those steps): (1) receiving computer data, (2) identifying a structure in the data, (3) linking to the identified structure potential actions that can be performed on the structure, (4) enabling the user to select the structure and a linked action to be performed on the structure, and (5) performing the selected action on the selected structure. A58(col.2:5-9).

The "Summary of the Invention" explains that when the system detects a structure, the system "links actions to the detected structure." A58(col.2:30-31).

"Each action is a computer subroutine that causes the CPU to perform a sequence of operations on the particular structure to which it is linked." A58(col.2:31-34). An action may "include internal actions, such as storing phone numbers in an electronic phone book, addresses in an electronic address book, appointments on an electronic calendar, and external actions such as returning phone calls, drafting letters, sending facsimile copies and e-mail, and the like." A58(col.2:36-41).

An embodiment disclosed in the specification describes an example of how the invention works. Figure 5 shows an example document, which has data with recognizable structures in it, including a phone number, an address, an e-mail address, and a name. A60(col.5:19-22).

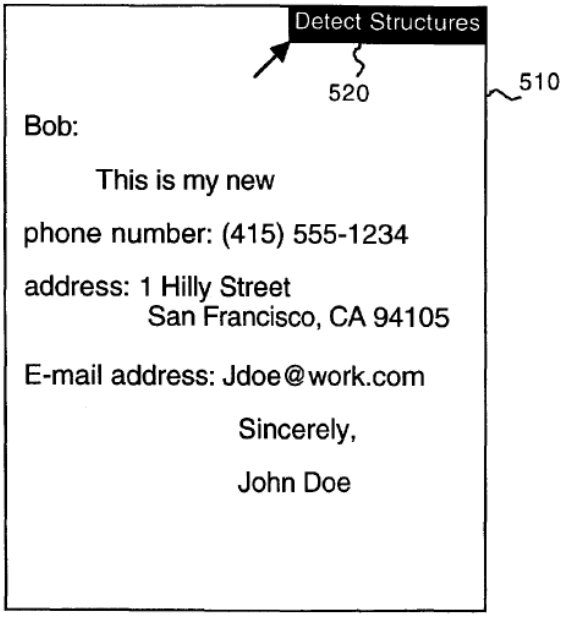


FIG. 5

A52.

The computer parses the contents of the document and identifies the telephone number, address, e-mail address, and name. A60(col.5:22-31). The computer makes the detected structures selectable, as depicted in Figure 6:

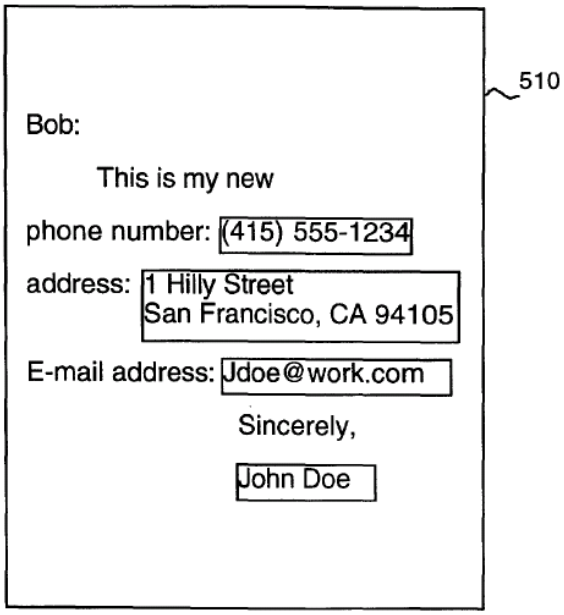


FIG. 6

A53, A60(col.5:29-37).

The computer "links" the appropriate "actions . . . to these identified structures"—in this example, the telephone numbers, addresses, e-mail addresses, and names. A60(col.5:31-33). For example, the computer may link to the detected telephone number the actions of "dialing the number and putting the number into an electronic telephone book." A60(col.5:40-43).

When the user selects a detected structure, the computer enables the user to select an action that the computer linked to that structure. Figure 7 below depicts

an example in which a user selects a detected telephone number. The computer displays for the user the options to "Call #" or "Put in electronic telephone book." A54. Those options are displayed because those are the actions that the computer linked to the detected telephone number. A60(col.5:31-43).

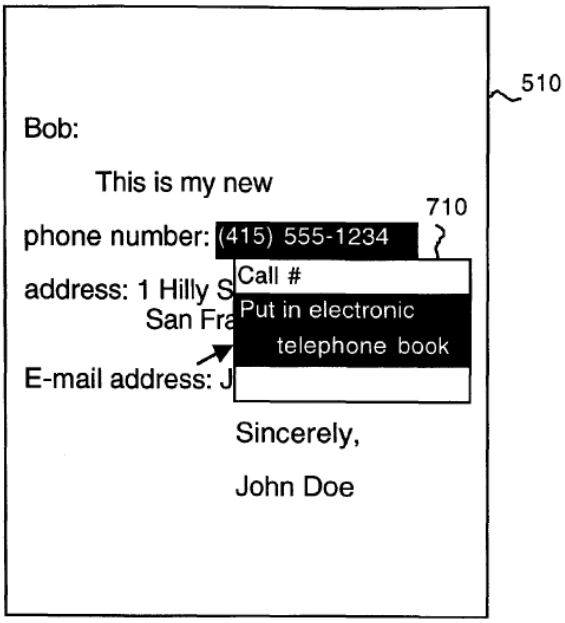


FIG. 7

A54.

If the user selects "the action for putting the number in an electronic telephone book," the computer "locates and opens the electronic telephone book, places the telephone number in the appropriate field and allows the user to input any additional information into the file." A60(col.5:44-50).

Figure 9 is a flowchart depicting a method for selecting and performing an action on a detected structure:

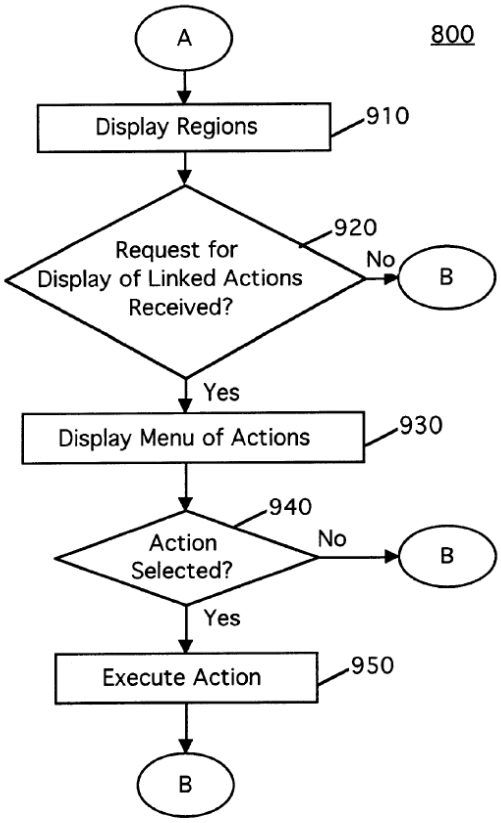


FIG. 9

A56. Once the computer detects structures in the data, the computer makes the detected structures able to be selected by the user (as indicated by boxes 910 and 920 in Figure 9) and links actions to the detected structures. A60(col.5:59-61, col.6:10-13). Once the user selects a detected structure (A60(col.6:16-20)), the computer "causes the candidate actions linked to the structure to be displayed" (as indicated by boxes 930 and 940). A60(col.6:20-21). For example, when the user selects a detected telephone number, the computer may display the linked actions of calling the detected number or putting the detected number in an electronic

telephone book. A54. When the user selects a linked action to be performed, then the action is executed on the detected structure that the user selected, as indicated by box 950. A60(col.6:28-29).

Each claim at issue relates to a method with five steps: (1) receiving computer data, (2) detecting a structure in the data, (3) linking at least one action to the detected structure, (4) enabling selection of the detected structure and also selection of a linked action, and (5) executing the selected action linked to the selected structure.

Independent claim 13 recites:

> 13. A program storage medium storing a computer program for causing a computer to perform the steps of:
> receiving computer data;
> detecting a structure in the data;
> linking at least one action to the detected structure;
> enabling selection of the structure and a linked action; and
> executing the selected action linked to the selected structure.

A61(col.8:1-9).

Independent claim 15 recites as follows:

> 15. In a computer having a memory storing actions, a method for causing the computer to perform an action on a structure identified in computer data, comprising the steps of:
> receiving computer data;
> detecting a structure in the data;
> linking at least one action to the detected structure;

> enabling selection of the structure and a linked action; and
>
> executing the selected action linked to the selected structure.

A61(col.8:22-33).

Claims 16, 17-20, 21, 23, and 24 depend from claim 15. A61-A62(col.8:34-50, col.8:51-58, col.10:1-4). In particular, claims 17, 18, and 19 recite differences relevant to the Board's (erroneous) conclusion that Nokia anticipates them:

> 17. The method recited in claim 15, wherein the memory contains *grammars*, and wherein the step of detecting a structure further comprises the steps of *retrieving a grammar and parsing the data based on the grammar*.
>
> 18. The method recited in claim 17, wherein the *grammar is associated with a particular action*, and wherein the step of linking at least one action to the detected structure includes the step of linking the particular action to the detected structure.
>
> 19. The method recited in claim 15, wherein the memory contains *strings*, and wherein the step of detecting a structure further comprises the steps of *retrieving a string from the memory and scanning the data to identify the string*.

A61(col.8:36-50) (emphasis added).

B. Proceedings Below

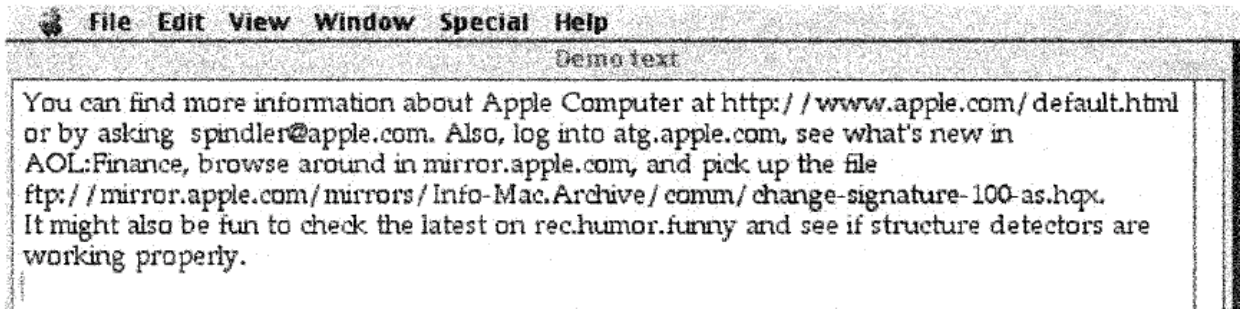
The PTO ordered *ex parte* reexamination of all the claims (i.e., claims 1-24) of the '647 patent. A1417-A1431 at A1419. The examiner initially rejected each claim in light of at least one of several purported prior-art references, including (1) U.S. Patent No. 5,859,636 (the "Pandit" reference), (2) a user manual titled

"Perspective Handbook" by Pensoft Corporation (the "Pensoft" reference), and (3) European Patent Office Publication No. 0 458 563 by Nokia Mobile Phones Ltd. (the "Nokia" reference). A1432-A1450.

1. Apple's reduction-to-practice evidence overcame the Pandit reference

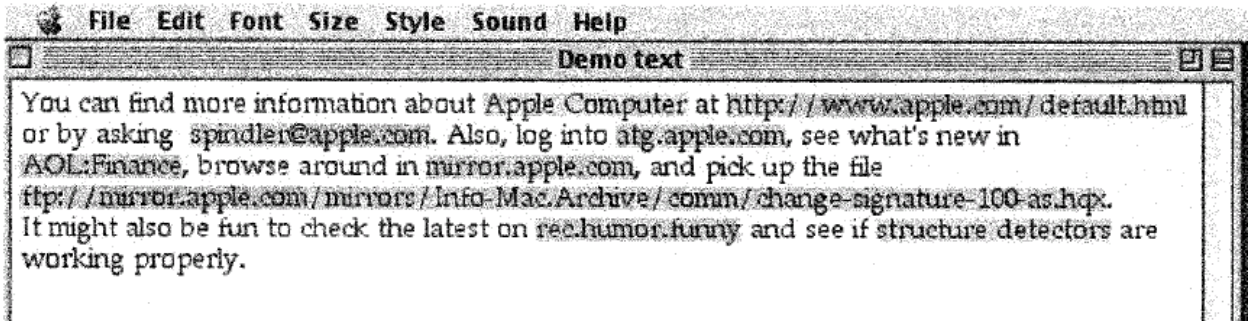
Apple responded to the initial rejection in light of Pandit, arguing that Pandit was not prior art because the invention of the '647 patent had been reduced to practice before Pandit's filing date. A1478-A1517; A2448-A2480 at A2452-A2454. Based on Apple's evidence, the examiner agreed and withdrew the rejections based on Pandit. A2454-A2455. The examiner thus confirmed claims 1-12 and 14, the computer-based system claims. A2476-A2477.

While Pandit is thus not at issue here, the evidence of reduction to practice shows another working example of the invention disclosed in the '647 patent. The screenshot below displays sample text (e.g., an e-mail) to which the invention may be applied:



A1504; *see* A2453 (discussing Exhibit B-2).

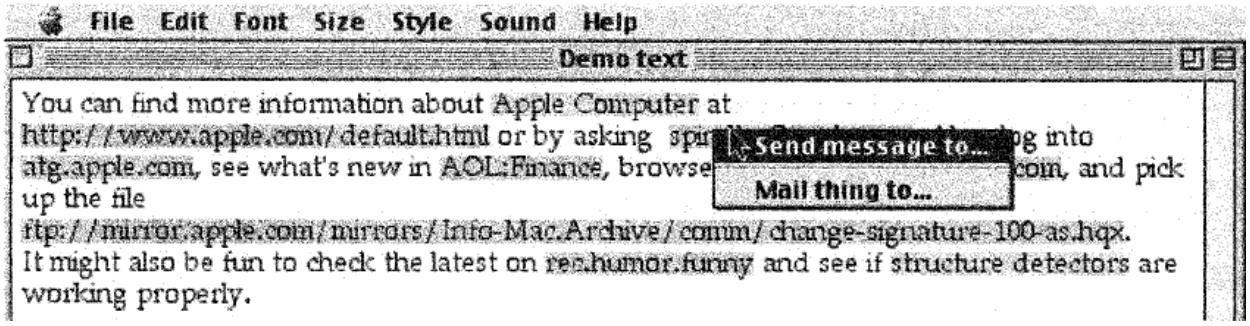
The computer detects structures in the data and highlights them for display to the user. In the screenshot below, the computer has detected structures such as a company name (Apple Computer), a website URL (http://www.apple.com/default.html), and an e-mail address (spindler@apple.com):



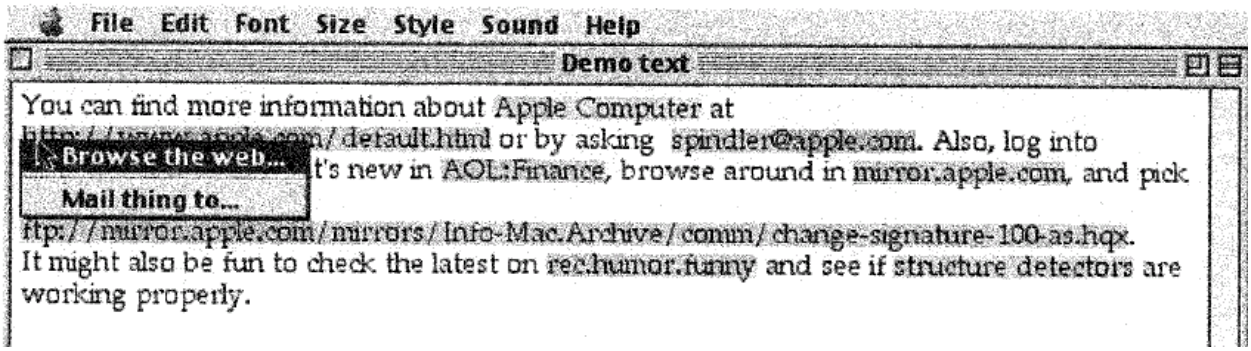
A1506; *see* A2453 (discussing Exhibit B-3).

Behind the scenes, the computer links potential actions to each detected structure. When the user selects a detected structure, the computer displays a "pop-up menu," which displays the "selectable actions related to" the structure. A2453.

For example, if the user selects an e-mail address, the computer offers the user two actions linked to the e-mail address: "Send message to..." and "Mail thing to...":



A1508; *see* A2453 (discussing Exhibit B-4). Similarly, if the user selects a website URL, the computer displays the actions it has linked to the URL:



A1510; *see* A2453 (discussing Exhibit B-5).

2. The Pensoft reference

The Pensoft reference is a user manual from 1992 for a software package called "Perspective." A76-A364. The Perspective software was a relational database used on personal digital assistants ("PDAs") running the PenPoint operating system, such as the EO Personal Communicator. A87; A1446; A2460. The Pensoft reference explains that Perspective helps users "manage your schedule, tasks, contacts, notes, and other important information." A87.

Information within Perspective was organized into categories, such as "People," "Companies," "Appointments," "To Do's," and "Notes." A92. Each category consists of several "details." A92. "For example, details for the Person category include name, business phone, and address." A92. Each occurrence of a category is called an "item." A93. Thus, each individual person is an item within the Person category. A93.

Information is displayed to the user in "documents." A94. Perspective includes six standard documents: "Day Planner," "Month Planner," "Address Book," "To Do List," "Topic Index," and "Note Index." A94. Each document has a set of items that it displays. A95. For example, the Address Book displays all the items in the People and Company categories. A95.

Items within the Perspective software could be linked together. A122. The Pensoft reference defined links: "A link represents a relationship between two items." A97, A122, A307. "For example, a person is linked to the company at which they work." A122. Thus, "[w]hen you look at a person's profile you see a link detail named Company. The content of the link is the name of the company where the person works." A122.

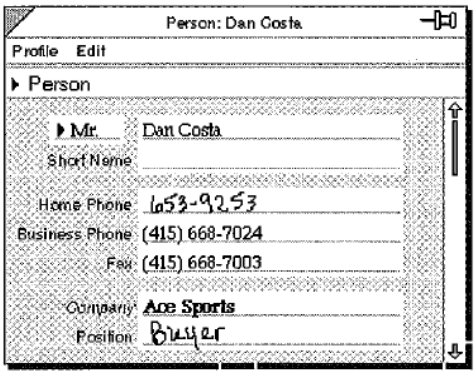
Links between items could be created manually or through a tool called the "Associate." A122-A123, A140-A141. The Associate monitored the data being input and tried to automatically create links between items of information:

> When the Associate is On, it looks at information you write in text, and whenever possible, creates a link. For example, when you enter an appointment with a person, the Associate looks to see if you have previously entered the person. If it finds a match, it creates a link between the appointment and the person's profile. Linking makes finding relevant information easier.

A123.

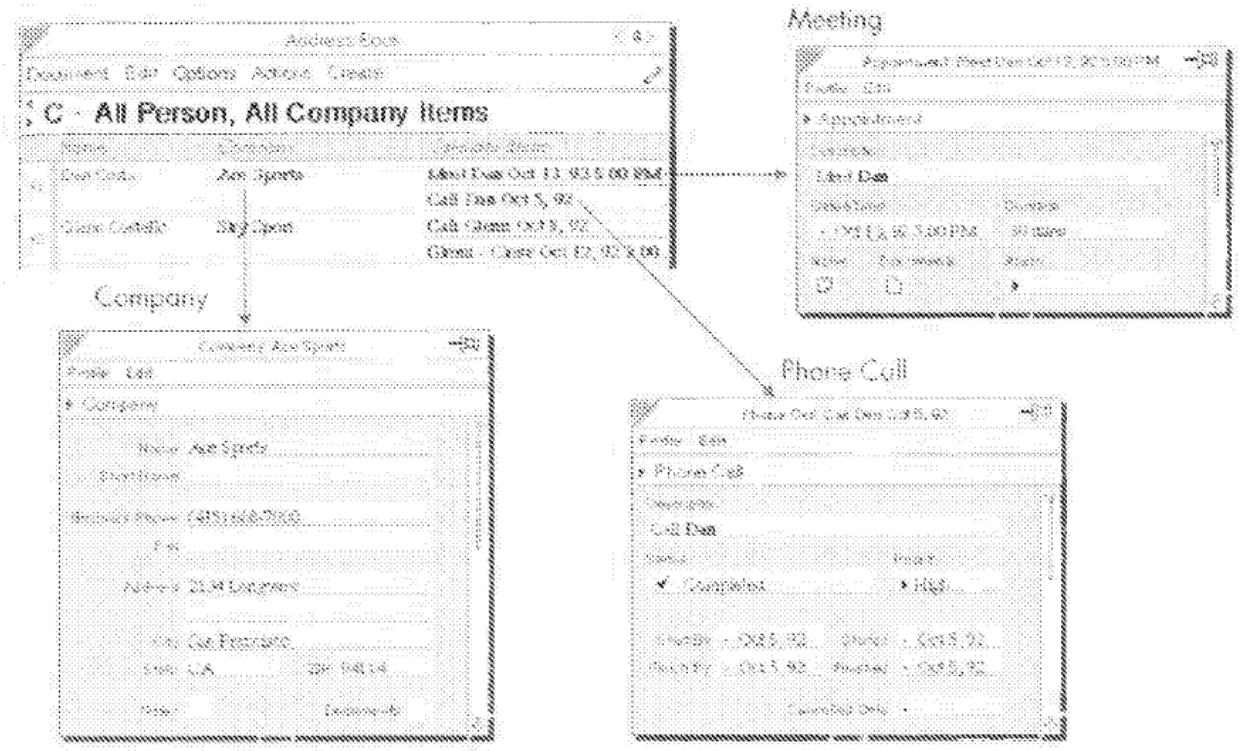
Throughout the Pensoft reference, links are defined expressly as mere relationships between two items of information, not as a relationship between an item and an action as in the ’647 patent. *See, e.g.*, A97, A122, A307 (“A link represents a relationship between two items.”); A98 (“establishing links between related items”); A129 (“any item that is linked to an item”); A140 (“create a link between two items”); A140 (“The link name states the relationship between the original item and the item to which you are linking.”); A141 (“Link an item in a document to another item”).

Each item in the Perspective software could be displayed in a “profile.” A95. “Profiles are forms which display all the details for an item.” A95. For example, the profile below displays the details for the “Dan Costa” item within the Person category:



A95. The user could “open a profile for any item displayed in a Perspective document.” A129.

The user also could "open a profile for any item that is linked to an item." A129. To open a profile for a linked item, the user would double-tap the linked information. A129. For example, when looking at the Address Book, which displayed information about people, if the user double-tapped on the name of a Company linked to the person, the profile for that Company would be displayed:



A97.

At most, therefore, Pensoft discloses a relational database with links between different items of information. Although the Associate tool in Perspective tries to detect items of information as the user enters them, and automatically links the detected item to already-entered items, that is not linking an action to the detected item as in the '647 patent. A123.

3. The Nokia reference

The Nokia reference is a European patent application publication from 1991 that was directed to a multifunction telephone. A1257-A1266. The device—referred to in the reference as "RPK"—was capable of receiving and displaying alphanumeric messages. A1257.

According to the Nokia reference, "messages received by RPKs will typically include telephone numbers to which the user of an RPK is requested to place calls." A1259(col.4:47-50). "Without this invention, the user would have to record the telephone numbers from the display either in his memory or on paper" and then "key the telephone number into his RPK." A1260(col.5:9-17). The Nokia reference explained that one of the primary advantages of the invention claimed in the patent was that it was "capable of interpreting as a telephone number a character string included in an arbitrary text message, either automatically or aided by the user," and was "capable of calling this number, when necessary." A1258(col.2:23-30).

The RPK could be used to place a telephone call to a number received in a text message by following three "principal steps." A1260(col.5:25-27).

First, the "RPK searches for a telephone number contained in a message." A1260(col.6:11-12). The identification of a telephone number could be performed either by the RPK or by the user. The RPK could "search[] from the message a

character string which could be a telephone number" based on "various criteria." A1260(col.6:23-25). Or the user could manually scroll through the message and then, "upon noticing a telephone number in the message, halt the display" so that the telephone number is displayed. A1260(col.6:37-38).

Second, the "user accepts the number." A1261(col.7:15). Accepting the number simply causes the telephone number to be displayed in the telephone's display:

> If the user accepts the number he performs an acceptance keying, whereupon the RPK will eliminate from the display any characters following the number and any non-numeric characters inside the number. Now there is visible in the display only a "pure" telephone number, and this number is also in the number register, according to which further procedures are possible.

A1261(col.7:32-39). If necessary, the "user may correct the telephone number by adding or eliminating digits at the beginning of the telephone number." A1261(col.7:40-42).

As a result of the first and second steps, "the register and the display of the telephone are now precisely the same as they would be upon the user having fed in the telephone number in its entirety and conventionally, by means of the number keys." A1261(col.8:6-10).

Third, the "RPK makes a call to the number." A1261(col.8:1). The telephone call is made not because the RPK linked an action to the telephone

number as in the ’647 patent, but rather as a result of conventionally pressing the call key: “A call to a number also takes place in the same manner as in a ‘normal’ call.” A1261(col.8:10-11).

Thus, at most Nokia merely discloses identifying a potential telephone number in a text message, displaying the number once the user accepts (and potentially corrects) it, and calling the number conventionally. In this regard, Nokia is no better than the prior art systems identified and criticized in the Background of the Invention section of the ’647 patent, which also could detect structures but required conventional means to perform actions on the detected structures. *See* A58(col.1:36-65).

4. Apple’s response and the examiner’s final rejection

Apple responded to the examiner’s initial rejection, arguing *inter alia* that neither Pensoft nor Nokia is anticipatory. A2570-A2605 at A2591-A2603.

In its response, Apple also proposed new claims 25-34. A2588-A2590. New claims 25-31 were method claims, each of which required nearly identical steps to those required by independent claims 13, 15, and 22. A2588-A2589. Claims 32-34 claimed computer-based systems with elements for performing the above steps. A2590.

In the final office action, the examiner determined that claims 1-12, 14, and 32-34 are patentable, but she rejected claims 13 and 15-31. A14.

5. The Board's decision

The Board affirmed the examiner's rejection of claims 13 and 15-31.

Pensoft. The Board held that claims 13, 15, 16, 20-22, and 24-31 are anticipated by Pensoft.

The Board concluded that Pensoft discloses a linked action, namely, the opening of a profile when the user selects an item of information. A4-A6. Although Pensoft expressly defines links as relationships between two items, not between an item and an action, the Board concluded that the purported action of opening a profile was linked to an item: "In Pensoft, the profile is 'linked' to the selected item because the specific profile that corresponds to the selected item (and not an unrelated profile) is launched upon selection of the item." A6.

Moreover, although the opening of a profile simply displays the details of a previously entered item of information, the Board concluded that the opening of a profile is a linked action: "Pensoft discloses that upon selection of an item in a document, the computer (or CPU) opens a corresponding profile. In order for the computer to perform such an operation, the computer would have executed a program (or subroutine) to do so." A5 (citation omitted). The Board concluded that "[e]ven assuming Appellant's contention to be correct that an 'action' must be a subroutine that causes the CPU to perform operations on a structure to which it is linked, Appellant has not persuasively demonstrated that the action of Pensoft (i.e.,

opening a profile) is not a subroutine that causes the CPU to perform operations." A5.

In the '647 patent, actions are linked to and performed on the structure that was *detected*. A58(col.2:30-34), A61(col.8:24-25, col.8:60-61); A2588-A2589. Yet the Board did not explain how opening a profile for an item—which necessarily occurs with respect to a previously entered item of information (A129)—is an action performed on a *detected* structure—which is by definition a separate, newly entered word or phrase recognized by the Associate tool (A123).

Nokia. Having affirmed the rejection of claims 13, 15, 16, 20-22, and 24-31 in light of Pensoft, the Board determined it did not need to reach the rejection of those claims over Nokia. A6. As to the claims it did reach, the Board held that claims 17-19 and 23 are anticipated by Nokia. A7-A10.

The Board determined that Nokia discloses linking an action to a detected structure. In particular, although linked actions in the '647 patent are actions performed by the *computer* (A58(col.2:31-34), A61(col.8:24-25)), the Board concluded that "the action of the *user* accepting the number" is linked to a detected telephone number. A7 (emphasis added).

The Board reasoned: "Nokia discloses that an apparatus (i.e., 'RPK') 'makes a call to [a telephone] number' (col. 5, l. 37) after the apparatus 'searches out' the telephone number in a message and 'the user accepts the number' (col. 5,

ll. 34-36)." A7 (alteration in original). "Nokia further discloses that after the user 'accepts the number' (col. 7, l. 15), the 'RPK makes a call to the number' (col. 8, l. 1) 'only when the number to be used is . . . in the display as a consequence of [the user accepting the number]' (col. 8, ll. 3-5)." A7 (alteration and omission in original). "The number is associated with (or 'linked') to the action of the user accepting the number because the apparatus makes a call to the number only 'as a consequence of' the user performing the action (i.e., accepting the number)." A7. Thus, although the Board appeared at one point to recognize that a link is an association or connection created by the computer (A6), it ultimately treated two events as necessarily linked just because one follows the other.

The Board also concluded that Nokia discloses enabling selection of the structure and a linked action because in Nokia the user "'accepts the [telephone] number' after the apparatus 'searches out' the telephone number in a message (col. 5, ll. 34-35) and the apparatus call[s] the telephone number 'as a consequence' (col. 8, ll. 4-5) of the user accepting the telephone number." A8 (first alteration by the Board).

Thus, the Board treated a single component—"the action of the user accepting the number"—as satisfying two separate steps of the patent. A7-A8. In particular, the Board determined that such purported action is linked to the detected structure in the claimed step of "linking at least one action to the detected

structure." A7. Yet it also concluded that that same action simultaneously meets the "selection of the structure" in the claimed step of "enabling selection of the structure and a linked action." A8. The Board so concluded even though those discrete steps must be performed separately in the claims at issue. A61(col.8:30-31).

With respect to claim 18, the Board determined that Nokia discloses a grammar associated with a particular action: "Nokia discloses a telephone number (or a 'grammar') that is 'associated with' an action (e.g., selecting the telephone number and calling the telephone number as a consequence of the selecting of the telephone number)." A9. Even though the patent distinguishes between a structure and a grammar used to detect the structure in the data, the Board conflated the two: "Appellant does not demonstrate a difference between the telephone number (or structure of the telephone number) and the 'grammar' as recited in claim 18." A9.

As to claim 19, the Board concluded that Nokia discloses that "the memory contains strings, and wherein the step of detecting a structure further comprises the steps of retrieving a string from the memory and scanning the data to identify the string." A10. The string method of searching taught in the '647 patent is that the computer memory stores strings (e.g., a list of important names) and searches the received data for a string (e.g., a single name) in the message identical to one of the

strings stored in memory. A60(col.5:14, col.6:44-46), A61(col.8:47-50). The board concluded that the string method in claim 19 was practiced by Nokia simply because Nokia tries to detect telephone numbers in a message: "Nokia discloses searching a message for a telephone number." A10. According to the Board, there are no "differences between searching a message for a telephone number and scanning data (or a message) to identify a string (or telephone number)." A10.

The Board also concluded that Nokia discloses a memory containing strings because "Nokia discloses that the apparatus 'stores the number under a name in the directory' (col. 8, ll. 47-48)," and the apparatus also stores in memory the "message that is searched." A10. The Board did not explain, however, how any of that information stored in memory in Nokia is used to search for a particular string in a received text message.

SUMMARY OF ARGUMENT

I. The Board erroneously concluded that claims 13, 15, 16, 20-22, and 24-31 of the '647 patent are anticipated by Pensoft for two reasons.

First, Pensoft does not disclose "linking at least one action to the detected structure." Pensoft describes the linking of two items of *information* together, not the linking of an *action* to a structure. Indeed, Pensoft expressly defines a link as a relationship between two items of information.

Second, Pensoft does not disclose executing a "linked action" on a structure. The Board concluded that the opening of a profile in Pensoft is a linked action. But that was based on an erroneous construction of "linked action." The claims and specification define a "linked action" as a computer subroutine that causes the CPU to perform a sequence of operations on the particular structure to which it is linked. The Board, while purporting to accept that construction *arguendo*, actually read "linked action" so broadly as to include merely opening a profile. Opening a profile simply displays the details of an item for the user; it does not perform a sequence of computer operations on the item.

Under the correct construction, Pensoft does not disclose a linked action. Pensoft states that the alleged action—opening a profile—merely opens a form that displays the details for an item. Even if opening a profile involves executing a computer subroutine, that does not mean the computer performs operations on the structure to which the purported action is linked.

Regardless of the construction of "linked action," opening a profile is not a linked action because it is performed not on the detected structure but on an entirely separate item of information.

Moreover, Pensoft does not disclose enabling selection of a linked action. The Pensoft system opens a profile automatically upon selection of a structure, without independently enabling the user to choose a linked action. Because

opening a profile is performed automatically, not selected by the user, it cannot be the linked action recited in the claims.

II.A. The Board erroneously concluded that claims 17-19 and 23 are anticipated by Nokia. At least three claim limitations are not disclosed in Nokia. Nokia therefore cannot anticipate those claims or any of the claims at issue here.

First, Nokia does not disclose "linking at least one action to the detected structure." In the '647 patent, that step is performed by the computer and is a separate step from the steps of "detecting a structure" and "enabling selection of a structure and a linked action." In Nokia, however, once the RPK detects a potential telephone number, no further processing is performed by the RPK until the user accepts (i.e., selects) the telephone number. Nokia is missing the "linking" step.

The Board's conclusion that Nokia links "the action of the user accepting the number" to the detected telephone number is wrong for several reasons. The Board's rationale conflates the "linking" step and the "enabling selection" step of the claims, allowing the acceptance of a number in Nokia to satisfy both limitations. The Board's reasoning also ignores that a linked action in the claims of the patent is a computer action, whereas the alleged action of the user accepting the number in Pensoft is a user action. And although the Board concluded that there must be a link because calling a number is the "consequence of" the user

accepting the number, that reasoning misstates how Nokia operates. It also ignores that links in the '647 patent claims are created within the computer and do not exist simply because one thing (accepting the number) may or may not potentially lead to another (calling the number).

Second, Nokia does not disclose "enabling selection of the structure and a linked action." Although Nokia discloses that the user can select the telephone number, it does not disclose that the RPK enables the user to select a linked action to be performed on that number. There simply is no reference to enabling the user to select an action linked to the telephone number, as the '647 patent teaches.

Third, Nokia does not disclose "executing the selected action linked to the selected structure." Because there is no action in Nokia that is linked to the selected structure, there is no execution of a linked action. Indeed, Nokia's RPK is an example of the prior art criticized in the patent as calling telephone numbers through merely conventional means.

B. Claim 18 should not have been rejected for an additional, independent reason: Nokia does not disclose the "grammar associated with a particular action" limitation. All that Nokia discloses is that a particular pattern is used to detect a potential telephone number. Even assuming that pattern is a grammar, Nokia does not state that any particular action is ever linked to that grammar.

C. Nokia also does not anticipate Claim 19 for another independent reason: Nokia does not disclose that the RPK's "memory contains strings" and that such strings are used in the step of "detecting a structure." The '647 patent teaches that the computer can search data for strings (e.g., names) identical to those in a library of strings (e.g., a list of names). The Board wrongly concluded that Nokia discloses this "string" method simply because Nokia discloses searching a message for a telephone number. But the mere fact that Nokia tries to detect a telephone number does not mean it uses strings to do so.

STANDARD OF REVIEW

This Court reviews only the grounds for rejection appearing in the Board's decision. *In re Thrift*, 298 F.3d 1357, 1367 (Fed. Cir. 2002).

"Although the PTO gives claims the broadest reasonable interpretation consistent with the written description, claim construction by the PTO is a question of law that [this Court] review[s] *de novo*, just as [the Court] review[s] claim construction by a district court." *In re Baker Hughes Inc.*, 215 F.3d 1297, 1301 (Fed. Cir. 2000) (citations omitted).

"Anticipation is a question of fact reviewed for substantial evidence." *In re Suitco Surface, Inc.*, 603 F.3d 1255, 1259 (Fed. Cir. 2010). "Substantial evidence is something less than the weight of the evidence but more than a mere scintilla of evidence and means such relevant evidence as a reasonable mind might accept as

adequate to support a conclusion." *Id.* (citations and internal quotation marks omitted).

ARGUMENT

A claim can be rejected as anticipated only if each and every element of the claimed invention is disclosed within the four corners of a single prior art reference. *In re NTP, Inc.*, 654 F.3d 1279, 1302 (Fed. Cir. 2011); *In re Skvorecz*, 580 F.3d 1262, 1266 (Fed. Cir. 2009). "'But disclosure of each element is not quite enough,'" because "the prior art reference [must] show the claimed invention arranged or combined in the same way as recited in the claim in order to anticipate." *Net MoneyIN, Inc. v. VeriSign, Inc.*, 545 F.3d 1359, 1371 (Fed. Cir. 2008) (quoting *Finisar Corp. v. DirecTV Grp., Inc.*, 523 F.3d 1323, 1334 (Fed. Cir. 2008)). Here, there are multiple, significant differences between Pensoft and the the '647 claims rejected in light of Pensoft, and between Nokia and the '647 claims rejected in light of Nokia. Each difference independently requires reversal. *Skvorecz*, 580 F.3d at 1268. The PTO's rejection should be reversed.

I. THE BOARD'S REJECTION IN LIGHT OF PENSOFT SHOULD BE REVERSED

This Court should reverse the PTO's rejection of claims 13, 15, 16, and 20-22 of the '647 patent, as well as newly added claims 24-31, as anticipated by the Pensoft reference, for two independent reasons. First, there is no substantial evidence that Pensoft's Perspective program performs "linking" an action to the

detected structure. Second, under the correct construction of "linked action," the opening of a profile in Pensoft is not a "linked action" at all.

A. Pensoft Does Not Disclose "Linking At Least One Action To The Detected Structure"

Each claim that was rejected as anticipated by Pensoft recites that the computer performs the step of "detecting a structure in the data," followed by the step of "linking at least one action to the detected structure." A61(col.8:5-6, col.8:29-30, col.8:66-67); *see* A2588-A2589. The "linking" step in the '647 patent is not disclosed in Pensoft. Pensoft discloses only the linking of *items* of information together in a relational database, not the linking of an *action* to a structure.

Pensoft repeatedly and expressly defines a "link" as associating pieces of information together: "A *link* represents a relationship between two items." A97; A122; A307. An "item" is not an action; rather, an "item" is a specific piece of information, such as a person. A93 ("Each occurrence of a category is an *item*. Dan, Donna, and John are each items in the Person category."); *see* A92 (defining "category"). Pensoft provides an example of a link, which is between two items of information: "For example, people are employed by a company, so there is a relationship between person and company." A97.

Indeed, each time that links are disclosed in Pensoft, they are relationships between items of information. *See, e.g.*, A98 ("establishing links between related

items"); A129 ("any item that is linked to an item"); A140 ("create a link between two items"); A140 ("The link name states the relationship between the original item and the item to which you are linking."); A141 ("Link an item in a document to another item").

In the '647 patent, the term used to refer to an item of information is "structure." A58(col.1:13-16) ("Much data that appears in a computer user's day-to-day activities contains recognizable structures that have semantic significance such as phone numbers, e-mail addresses, post-office addresses, zip codes and dates."). Thus, at most, Pensoft discloses linking structures together.

But each rejected claim of the '647 patent requires a link between "at least one action" and "the detected structure." A61-62(col.8:6, col.8:30, col.8:67); A2588-A2589. Pensoft discloses no link between any action and a structure, as recited by the claims. Put simply, Pensoft at most discloses linking together two items or things (such as a person and a company), whereas the '647 patent claims require an altogether different type of linking—i.e., between a structure (or thing) and an *action* (such as storing, dialing, or e-mailing).

That alone is sufficient to reverse the Board's conclusion as to the claims rejected in light of Pensoft. "Anticipation cannot be found, as a matter of law, if any claimed element or limitation is not present in the reference." *Skvorecz*, 580 F.3d at 1268.

B. Pensoft Does Not Disclose Executing A "Linked Action" On A Structure

The rejections in light of Pensoft also should be reversed for another separate reason: the opening of a profile in Pensoft is not a "linked action." Each rejected claim contains three limitations requiring a linked action: "linking at least one action to the detected structure," "enabling selection of the structure and a linked action," and "executing the selected action linked to the selected structure." A61(col.8:5-9, col.8:30-33); *see* A61-A62(col.8:67-col.9:7); A2588-A2589. Because there is no linked action in Pensoft, it cannot anticipate.

1. The Board erroneously construed "linked action"

As Apple contended before the Board, a "linked action" is defined in the '647 patent as "a computer subroutine that causes the CPU to perform a sequence of operations on the particular structure to which it is linked." A5798 (quoting A58(col.2:31-34)). The "particular structure" on which the sequence of operations is performed is the "detected structure" to which the action was linked. A58(col.2:30-32) ("Upon detection of a structure, the analyzer server links actions to the detected structure. Each action is a computer subroutine . . . .").

Although the Board purported to accept *arguendo* that construction, the Board actually applied a different, unreasonably broad construction that is not supported by the claim language or the specification. Although the Board must give a claim term its broadest reasonable construction, "this court has instructed

that any such construction be 'consistent with the specification, . . . and that claim language should be read in light of the specification as it would be interpreted by one of ordinary skill in the art.'" *Suitco Surface*, 603 F.3d at 1260 (quoting *In re Bond*, 910 F.2d 831, 833 (Fed. Cir. 1990)).

The language of the rejected claims indicates that a linked action is a computer operation that is performed on a structure. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1314 (Fed. Cir. 2005) (en banc) ("the claims themselves provide substantial guidance as to the meaning of particular claim terms"). Claim 15, and thus the claims that depend from it, as well as all the newly added claims, provide that the claimed method "caus[es] the computer to perform an action on a structure identified in computer data." A61(col.8:24-25); A2588-A2589. Similarly, claim 22 provides that the claimed method is a "computer-based method for causing a computer to identify, select and perform an action on a structure in computer data." A61(col.8:59-61). Moreover, the claims provide that once the user selects a structure and a linked action, the linked action is "execut[ed]" on the detected structure. A61(col.8:9-10, col.8:32-33).

Consistent with the claim language, the "Summary of the Invention" in the specification expressly defines a linked action as a sequence of computer operations performed on the structure: "Upon detection of a structure, the analyzer server links actions to the detected structure. Each action is a computer subroutine

that causes the CPU to perform a sequence of operations on the particular structure to which it is linked." A58(col.2:30-34).

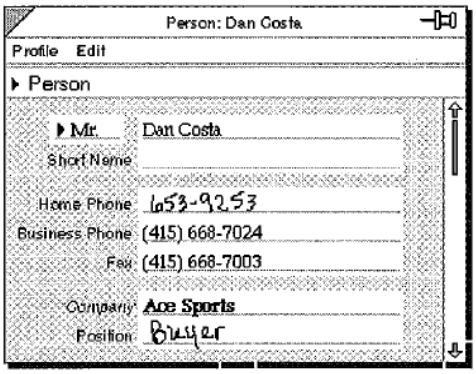
Moreover, all the examples of linked actions disclosed in the specification involve computer operations performed on the structure. The specification provides that "[a]n action may further include internal actions, such as storing phone numbers in an electronic phone book, addresses in an electronic address book, appointments on an electronic calendar, and external actions such as returning phone calls, drafting letters, sending facsimile copies and e-mail, and the like." A58(col.2:36-41). A linked action also may involve loading the detected structure into a different software program: "An action may specify opening another application, loading the identified structure into an appropriate field, and closing the application." A58(col.2:34-36).

The Board, however, read "linked action" so broadly as to include merely opening a profile. A5. But merely opening a profile does not perform a sequence of computer operations on the detected structure. Opening a profile simply displays the details of a selected item for the user. A95, A129. It does not perform any operations on a structure, such as changing the structure, storing the structure anywhere, loading the structure in another software application, or using the structure in any way. Nothing in the claim language or specification supports the Board's overly broad construction.

2. Pensoft does not disclose a linked action performed on a structure

Opening a profile in Pensoft is not a linked action for at least three reasons.

First, under the correct construction, opening a profile is not a linked action because it does not involve performing a sequence of operations on the detected structure. According to Pensoft, the user can open a profile to view the details of an item. A129 ("You can open a profile for any item displayed in a Perspective document."). Opening a profile in Pensoft merely opens a "form[] that display[s] all the details for an item." A129. An example opened profile is depicted below:



A95. *After* a profile is opened, the user can "enter or change any information" displayed in the profile. A129.

Opening a profile in Pensoft is not a linked action. As described above, a linked action is a computer subroutine in which the computer performs a sequence of operations (such as storing, sending, loading, etc.) after the action has been selected by the user on the particular structure to which it is linked. The Board reasoned that for a profile to be opened, the computer "would have executed a

program (or subroutine)." A5; *see* A5 ("'opening' a profile would have entailed an action at least because some activity would be performed to execute the operation"). Even if the computer executes a subroutine to show the profile, that does not mean that the computer performs a sequence of operations on the structure to which the action is linked. In the example from the figure above, opening the profile for "Dan Costa" does not mean that the computer performs a sequence of operations (storing, sending, loading, etc.) on "Dan Costa." Rather, opening a profile for "Dan Costa" merely displays a form showing all the details for Dan Costa. A129.

Second, regardless of the construction of "linked action," opening a profile is not a linked action because it is not performed on the "detected structure." In the '647 patent, a linked action is performed on the particular detected structure to which the action was linked. A61(col.8:23-24, col.8:59-60); *see* A58(col.2:30-34) ("Upon detection of a structure, the analyzer server links actions to the detected structure. Each action is a computer subroutine that causes the CPU to perform a sequence of operations on the particular structure to which it is linked."). For example, if the invention of the '647 patent detects a telephone number, the linked actions are actions to be performed on the telephone number, such as calling the number. A60(col.5:40-43).

That is not what happens in Pensoft. Detection in Pensoft's Perspective program is performed by a tool called the Associate. According to Pensoft, "[w]hen the Associate is On, it looks at information you write in text, and whenever possible, creates a link." A123. "For example, when you enter an appointment with a person, the Associate looks to see if you have previously entered the person." A123. If the Associate finds a match, then it creates a link to the matched item. Thus, if the user enters "Follow up with Dan" in the To Do list, the Associate may detect the word "Dan" in that entry and link it to the previously entered item for "Daniel Costa." A126. If the user opens a profile for that person, however, no action is performed on the detected structure "Dan" in "Follow up with Dan." Rather, opening a profile is performed on the item that the Associate linked to "Dan," the previously entered item for the person named "Daniel Costa." That is, clicking on "Dan" does not do anything with "Dan," the detected structure; rather, it opens the profile of a previously entered item.

Third, Pensoft does not disclose "enabling selection of the [detected] structure *and* a linked action." A61(col.8:7, col.8:31) (emphasis added); *see* A62(col.9:3-4); A2588-A2589. The Pensoft system opens a profile automatically upon selection of an item. Because opening a profile is not an action selected by the user, it cannot be the linked action recited in the claims.

The claims at issue require not only that the computer enable selection of a detected structure, but also that the user be able to select a linked action to perform on the detected structure. Selection of a detected structure and selection of a linked action are performed independently. They cannot be met by simply enabling selection of a detected structure, followed automatically by performance of an action on that structure, without having enabled the user to select a linked action to be performed.

Thus, for example, Figure 7 depicts that once the user selects a detected structure (a telephone number), the computer presents for user selection the actions linked to the telephone number:

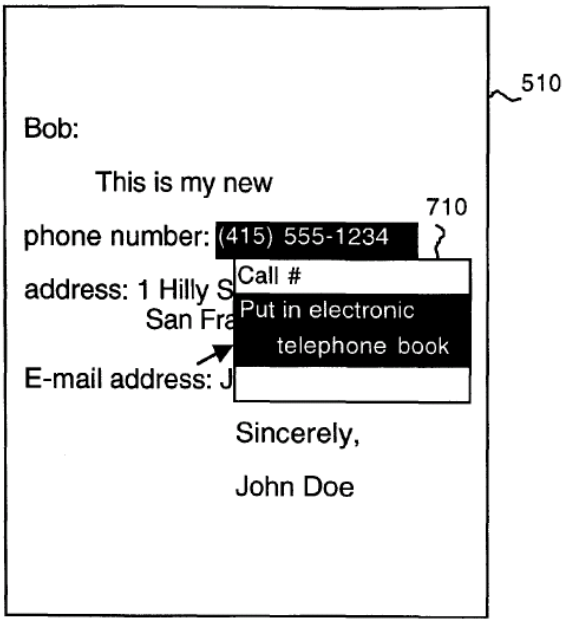


FIG. 7

A54. Enabling selection of the telephone number and enabling selection of the linked action are performed independently.

Likewise, the flowchart in Figure 9 of the ’647 patent shows enabling selection of the detected structure and enabling selection of the linked action as being independent:

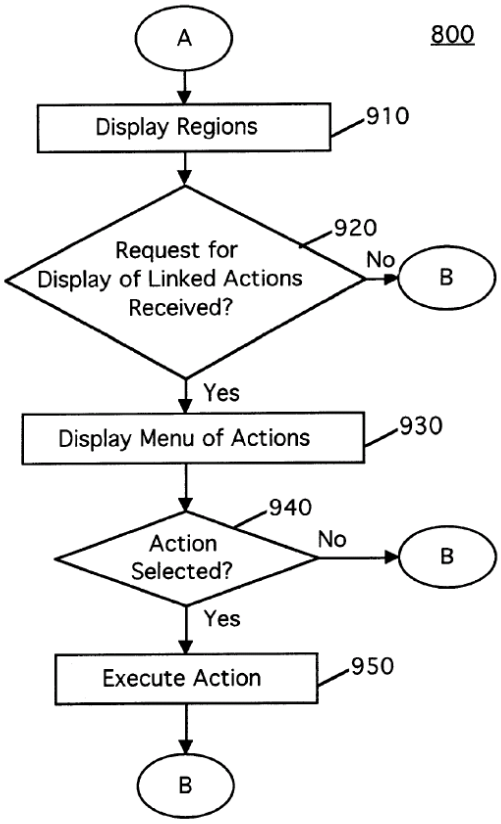


FIG. 9

A56. In boxes 910 and 920, the computer displays the structures that were detected and enables the user to select a structure—i.e., to request that the candidate linked actions be displayed. Once the user selects a structure, boxes 930

and 940 show that the computer presents the user with a menu of candidate actions and allows the user to select a linked action from the candidate actions.

Pensoft, by contrast, does not enable the user to select a linked action in addition to selecting a detected structure. In Pensoft, when the user simply double taps on information displayed in a document—i.e., when the user selects the information—the system automatically opens a profile for the item linked to that information. A129. The Pensoft system does not enable the user to choose a linked action to be performed on the selected structure. Automatically executing an action once the user selects a *structure* is not "enabling selection of . . . a linked *action*" as required by the claims. A61(col.8:7, col.8:31) (emphasis added); *see* A62(col.9:3-4); A2588-A2589. Because opening a profile is performed automatically and is not selected by the user, it cannot be the linked action recited in the claims.

Pensoft's lack of a linked action provides another independent reason for setting aside the Board's decision. The Board's decision affirming rejection of claims of the '647 patent as anticipated by Pensoft should be reversed.

II. THE BOARD'S REJECTION IN LIGHT OF THE NOKIA REFERENCE SHOULD BE REVERSED

A. Claims 17-19 And 23 Should Not Have Been Rejected As Anticipated By Nokia

The Nokia reference discloses a sequence of three steps for detecting and dialing a telephone number. Those steps are fundamentally different from the particular steps recited in claims 17-19 and 23 of the '647 patent for performing a linked action on a detected structure. Indeed, at a minimum, three limitations of the claimed methods are not disclosed by Nokia: (1) linking at least one action to the detected structure, (2) enabling selection of the structure and a linked action, and (3) executing the selected action linked to the selected structure. Failure to disclose any one of these limitations means that Nokia does not anticipate the rejected claims. *Skvorecz*, 580 F.3d at 1268.

1. Nokia does not disclose "linking at least one action to the detected structure"

In claims 17-19 and 23 of the '647 patent, once the computer receives data and detects a structure in the data, the computer "link[s] at least one action to the detected structure." A61(col.8:30). The Board interpreted "link" as "associated or connected with." A6; *see* A58(col.1:67). That is not disclosed in Nokia.

Linking must occur within the computer. All the steps are performed "[i]n a computer having a memory," and the claims are "method[s] for causing the computer to perform an action on a structure identified in computer data."

A61(col.8:23-25). Thus, linking an action to a detected structure means creating an association or connection within the computer between (1) the detected structure in the received data and (2) candidate computer subroutines that, upon execution, cause the CPU to perform a sequence of operations on the structure. A58(col.1:67, col.2:31-34).

The specification discloses at least one particular way that a link may be created. For example, in one non-limiting embodiment of the invention, the computer "links actions . . . to the detected structure, using conventional pointers," which "direct the system to the associated actions." A59-A60(col.3:65-67, col.5:2-3).

No linking of an action to a detected telephone number is disclosed anywhere in the Nokia reference. According to Nokia, after the RPK receives a text message and the user presses a particular key while the message is displayed, the RPK tries to detect a telephone number in the message. A1260(col.6:11-30). The RPK searches the message for the first point that could be a telephone number according to a particular defined pattern. A1260(col.6:22-25). If the RPK detects a potential telephone number, the display "halt[s]" at the point at which the potential telephone number is located in the message. A1260(col.6:31).

Once the RPK performs the step of detecting a potential telephone number, no more processing is performed by the RPK until the user takes further action.

The very next step after the RPK detects a potential telephone number is that the user can choose to accept (i.e., select) the telephone number. A1261(col.7:15-35).

That is, the process disclosed in Nokia goes from the step of detecting a structure to the step of enabling the user to select the structure. But in the '647 patent claims at issue, in addition to the steps of detecting a structure and enabling selection of the structure, the claims require the step of "linking at least one action to the detected structure." A61(col.8:30). That linking step is not disclosed in the Nokia reference. Nokia therefore cannot be anticipatory. *See Skvorecz*, 580 F.3d at 1268.

Despite the lack of any express disclosure of linking in Nokia, the Board concluded that Nokia links "the action of the user accepting the number" to the detected telephone number. A7. The Board reasoned that Nokia disclosed linking because "after the user 'accepts the number,' the 'RPK makes a call to the number' 'only when the number to be used is . . . in the display as a consequence of [the user accepting the number].'" A7 (quoting A1261(col.7:15, col.8:1-5)) (omission and alteration by Board). The Board concluded that the "number is associated with (or 'linked') to the action of the user accepting the number because the apparatus makes a call to the number only 'as a consequence of' the user performing the action (i.e., accepting the number)." A7.

That reasoning is flawed for at least four reasons. First, the Board erroneously relied on a single component as meeting two separate steps in the claims of the ’647 patent. The claims at issue make clear that “linking at least one action to the detected structure” and “enabling selection of the structure and a linked action” are performed separately in the claims at issue. A61(col.8:30-31). But the Board treated a single component—“the action of the user accepting the number”—as both being “link[ed] . . . to the detected structure” in the “linking” step and also as satisfying the user’s “selection of the structure” in the “enabling selection” step. A7-A8. Acceptance of the telephone number cannot be relied on to meet both limitations. *In re Robertson*, 169 F.3d 743, 745 (Fed. Cir. 1999); *see Lantech, Inc. v. Keip Mach. Co.*, 32 F.3d 542, 547 (Fed. Cir. 1994). Thus, even under the Board’s rationale, Nokia does not anticipate because it does not disclose all the elements “arranged or combined in the same way as in the claim.” *Net MoneyIN*, 545 F.3d at 1370; *Abbott Labs. v. Sandoz, Inc.*, 544 F.3d 1341, 1345 (Fed. Cir. 2008) (elements must be disclosed “in the same form and order as in the claim”).

Second, even assuming that “the action of the *user* accepting the number” is linked to the detected telephone number (A7 (emphasis added)), that does not read on the claims of the ’647 patent. The claims expressly recite that the action linked to and executed on the structure is performed by a computer. A61(col.8:24-25)

("method for causing the computer to perform an action on a structure identified in computer data"). But the "action" the Board determined was linked to the telephone number in Nokia is an action performed by the user, not the computer. A7.

Third, contrary to the premise of the Board's decision, the user's acceptance of a number in Nokia does not result in the number being dialed. Rather, when a user accepts the number, "the RPK will eliminate from the display any characters following the number and any non-numeric characters inside the number" so that "only a 'pure' telephone number" is displayed. A1261(col.7:33-38). Thus, as Nokia expressly states, the "consequence of" the user's acceptance of the number is simply that "the number to be used is in the register and in the display." A1261(col.8:3-5). Once the telephone number is in the display, the phone works conventionally: the user still must call the number "in the same manner as in a 'normal' call" by pressing a "call keying." A1261(col.8:6-15). The fact that the number is called only as a result of conventional dialing demonstrates that the telephone itself has not linked any particular action to the number.

Indeed, the Background of the Invention section of the '647 patent specifically criticizes prior art systems such as Nokia's RPK that "involve[d] detecting telephone numbers" and that "enable[d] a user to select a telephone number" but that simply dialed the number using conventional dialing.

A58(col.1:52-65). Such systems do not employ the advancements over the prior art that the '647 patent added, including linking actions to a structure and enabling selection of a linked action to be performed on the structure, and thus should not be viewed as reading on the patent. *Saffran v. Johnson & Johnson*, 712 F.3d 549, 560 (Fed. Cir. 2013) (reading claims as not encompassing prior art criticized in specification).

Fourth, the fact that an action is potentially the "consequence of" selecting a structure is not what the '647 patent means by linking an action to a detected structure. Linking within the patent refers to the computer creating a computer-based association or connection between a structure and at least one candidate action. A58(col.1:67). That is, the computer creates an association between the identified structure and the candidate computer subroutines that may be executed on the structure. The Board did not determine (nor could it) that the RPK creates any such association within the telephone between a detected telephone number and any computer subroutines. Nokia simply discloses that once the RPK detects a potential telephone number, the user potentially may accept the telephone number to put it into the display, and then the user potentially may press the call key while the number is in the display to make the call. There is no disclosure that the RPK creates a link after detecting a telephone number.

2. Nokia does not disclose "enabling selection of the structure and a linked action"

In Nokia, the result of the user accepting a number is that the display contains "only a 'pure' telephone number, and this number is also in the number register." A1261(col.7:36-38). That is, "the register and the display of the telephone are now precisely the same as they would be upon the user having fed in the telephone number in its entirety and conventionally, by means of the number keys." A1261(col.8:6-10). That does not disclose "enabling selection of the structure and a linked action" as required by the claims of the '647 patent. A61(col.8:31).

In the '647 patent claims at issue, once a structure is detected, the computer links an action to the structure and "enabl[es] selection of the structure and a linked action." A61(col.8:31). That is, the computer provides a way for the user to select a structure from among the one or more structures detected in the data, and also to select from among the one or more candidate actions linked to that structure.

The specification teaches that detected "structures are stored in memory and presented to the user for selection." A58(col.2:58-59). Once the user selects a structure, "a menu of candidate actions is presented, each of which may be selected and performed on the selected structure." A58(col.2:60-62).

In one embodiment, this "enabling selection" step is performed when the "[u]ser interface 240 . . . highlights the identified structures in document 210, and

makes the identified structures mouse-sensitive." A60(col.5:35-37). An exemplary user interface with highlighted structures is depicted in Figure 6:

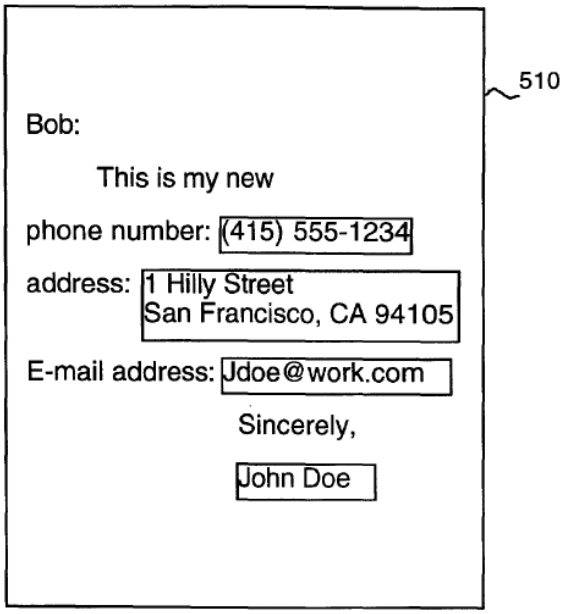


FIG. 6

A53.

According to this embodiment, once the user selects a structure, "user interface 240 presents a pop-up menu 710." A60(col.5:38-40). The "pop-up menu 710 displays the candidate actions linked to the selected telephone number grammar 410, including dialing the number and putting the number into an electronic telephone book." A60(col.5:40-43). The user can select from among the candidate actions to be performed on the telephone number. The example pop-up menu is depicted in Figure 7:

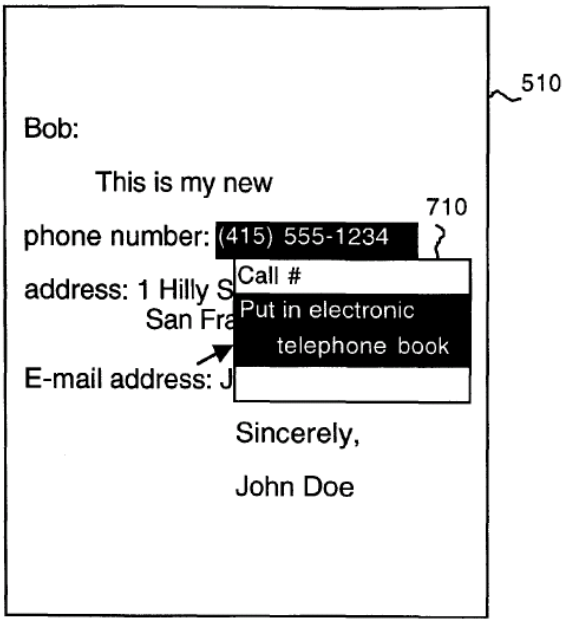


FIG. 7

A54.

The patent states that enabling selection is not limited to a visual interface but could also be performed through an audio interface or "a combination of sensory mechanisms." A59(col.4:32-57). "In the audio environment, user interface 240 may provide a special sound after application 167 reads a recognized pattern, and enable selection of the pattern through the use of an audio interface action, such as a voice command or the pressing of a button on the touch-tone telephone keypad as before. Thus, user interface 240 may present the linked actions via voice synthesis." A59(col.4:44-50).

The Board concluded that the Nokia reference discloses the step of enabling the selection of the structure and a linked action. A8. But the Board's own

explanation belies its conclusion. The Board observed that "Nokia discloses a user who 'accepts the [telephone] number.'" A8 (alteration in original) (quoting A1260(col.5:34-35)). The Board reasoned that "in order for a user to be able to 'accept' a telephone number on an apparatus . . . , the apparatus would enable the user to do so." A8. But the user simply accepting a telephone number is at most selecting the detected *structure*. That alone is not enough: the claims recite that the computer enables the user to select both the detected structure *and* a linked action from among the one or more candidate actions linked to the structure. A61(col.8:31).

Nothing in Nokia discloses that the RPK enables the selection of a linked action. To be sure, Nokia states that once the telephone number is in the display, the user might dial the number in a conventional manner using a "call keying." A1261(col.8:14). But the call key would operate on whatever is in the telephone's display. Unlike in the '647 patent, the user's ability to call the number using the call key does not result from the RPK having linked an action to the detected telephone number and having enabled the user to select the action from among the candidate actions linked to the telephone number. Nokia is simply devoid of any disclosure of a video or audio display, or any other means, that presents the actions linked to the telephone number and allows the user to select one, as the '647 patent teaches.

3. Nokia does not disclose "executing the selected action linked to the selected structure"

In the '647 patent, once the structure and linked action are selected, the computer "execut[es] the selected action linked to the selected structure." A61(col.8:32-33); *see* A58(col.2:51-53) ("When a candidate action is selected, the action processor performs the selected action on the selected structure."). For example, in one non-limiting embodiment, if the user selects the action of storing the telephone number in the electronic telephone book, then the computer "locates and opens the electronic telephone book, places the telephone number in the appropriate field and allows the user to input any additional information into the file." A60(col.5:47-50). Nokia does not disclose that "executing" step.

Indeed, because there is no action in Nokia that is linked to the selected structure, there is no execution of a linked action. According to Nokia, the actual dialing of the telephone number is not performed through execution of any linked action but rather through the user placing the call "in the same manner as in a 'normal' call." A1261(col.8:10-11).

Far from anticipating, Nokia is an example of the "[c]onventional systems" discussed in the "Background of the Invention" section of the '647 patent. A58(col.1:36). The '647 specification explains that conventional systems were able to "identify structures in computer data." A58(col.1:36-37). The problem with the conventional systems, which the '647 patent solved, was that those

systems did "not enable automatic performance of an action on an identified structure." A58(col.1:37-38). So too with Nokia: although Nokia is able to detect potential telephone numbers in a text message, Nokia does not enable automatic performance of a linked action on the telephone number.

* * *

Accordingly, the Board's rejection of claims 17-19 and 23 as anticipated by Nokia should be reversed. Moreover, if this Court reverses the rejection of claims 13, 15, 20-21, and 25-31 over Pensoft (as it should, *see supra* Part I), those claims are not anticipated by Nokia either, for the same reasons discussed above. The examiner and Apple treated claims 13, 15, 17-23, and 25-31 as a single group that would stand or fall together. A36-A37; A5801; *see* 37 C.F.R. § 41.37(c)(1)(iv) (claims that are grouped together as "subject to the ground of rejection" all "stand or fall together").

B. Claim 18 Should Not Have Been Rejected Because Nokia Does Not Read On The "Grammar Is Associated With A Particular Action" Limitation

In the event that the Court does not reverse the rejection of claim 18 for the above reasons, that rejection should be reversed for an additional, independent reason: Nokia does not disclose the limitation that recites that the "grammar is associated with a particular action." A61(col.8:41-45).

Claim 18 depends from claim 17, which in turn depends from independent claim 15. Claim 17 adds particular steps to the limitation of "detecting a structure in the data." Specifically, claim 17 recites that "the memory contains grammars" and that "the step of detecting a structure further comprises the steps of *retrieving a grammar* and *parsing the data based on the grammar*." A61(col.8:37-40) (emphasis added). A grammar is a particular type of data pattern that can be used to recognize information in a document, such as dates, addresses, phone numbers, and names. A58-A59(col.1:27-31, col.3:61-64). The specification explains that when grammars are used in the invention, the "[p]arser 310 retrieves a grammar from grammar file 320 and parses text using the retrieved grammar." A59(col.4:62-64).

Claim 18 adds a grammar-related limitation to the step of "linking at least one action to the detected structure." In claim 18, each grammar is associated with a particular action. When a particular grammar is used to detect a structure in the data, the action that is associated with that grammar is the action that gets linked to the detected structure. Claim 18 thus recites that "the grammar is associated with a particular action," and "the step of linking at least one action to the detected structure includes the step of linking the particular action to the detected structure." A61(col.8:41-45).

Figure 4 of the ’647 patent depicts the use of grammars according to claim 18. A60(col.5:6-7). Figure 4 shows grammars that can be used to detect different types of structures, as well as actions associated with the grammars. For example, “[o]ne of the grammars 410 is a telephone number grammar with associated actions for dialing a number identified by the telephone number grammar or placing the number in an electronic telephone book.” A60(col.5:8-12). Another grammar is an e-mail address grammar, which is associated with the actions of sending an e-mail or putting the e-mail address in the address book:

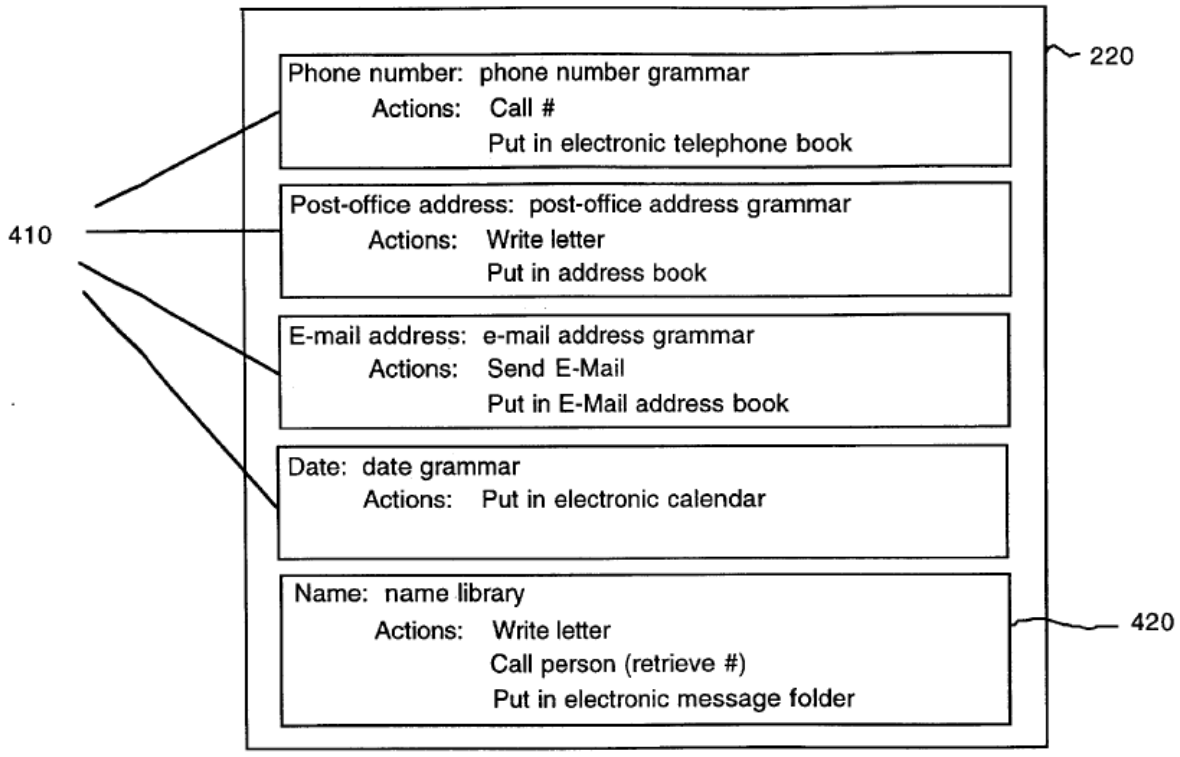


FIG. 4

A51.

When a grammar is used to detect a structure, the computer takes the actions that are associated with the grammar and links those actions to the detected structure. A61(col.8:41-45). In the Figure 4 example, when the computer "identifies an address using the 'e-mail address' grammar, actions for sending e-mail to the identified address and putting the identified address in an e-mail address book," i.e., the actions associated with the grammar, "are linked to the address." A60(col.5:15-18).

For example, in one non-limiting embodiment, the way that the grammar initially is associated with actions is through the use of conventional pointers. A59-A60(col.4:67-col.5:1). These pointers direct the system to the associated actions. A60(col.5:2-3). In this embodiment, when a structure is identified through the use of a grammar, the computer "retrieves from grammar file 320 pointers attached to the grammar and attaches the same pointers to the identified structure." A59-A60(col.4:64-col.5:2).

The RPK disclosed in the Nokia reference does not perform the method recited in claim 18 and therefore cannot anticipate it. Nothing in Nokia states that the way an action purportedly is linked to the detected telephone number is through first associating that action with a grammar used to detect the telephone number.

All that Nokia discloses regarding the detection of a telephone number is that a particular pattern is used:

> [T]he RPK will search, starting from the beginning of the message, for the first point at which there is, for example, a numeral and the numeral is followed by at least four of any of the characters included in the following set: numerals, -, /, (,), space. At least three of these first five characters must be numerals.

A1260(col.6:16-21).

Even assuming that this particular pattern is a grammar for detecting a telephone number, Nokia does not state that any particular action is ever associated with the grammar itself. But that is required by claim 18. A61(col.8:41-42) ("the grammar is associated with a particular action").

Nor does Nokia provide that "the particular action" purportedly associated with the telephone-number grammar is linked to a detected telephone number. That too is required by claim 18. A61(col.8:44-45).

The Board's conclusion that Nokia reads on claim 18 is based on its faulty view that a telephone number is the same as a grammar used to detect a telephone number. The Board reasoned that claim 18 is anticipated because "Nokia discloses a telephone number (or a 'grammar') that is 'associated with' an action (e.g., selecting the telephone number and calling the telephone number as a consequence of the selecting of the telephone number)." A9. The Board did not see any "difference between the telephone number (or structure of the telephone number) and the 'grammar' as recited in claim 18." A9.

The Board's rationale cannot be squared with the '647 patent, which expressly provides that a grammar and a structure are distinct. *Bicon, Inc. v. Straumann Co.*, 441 F.3d 945, 950 (Fed. Cir. 2006) ("claims are interpreted with an eye toward giving effect to all terms in the claim"). According to claim 17, a grammar is used in "the step of detecting a structure." A61(col.8:38-40). The specification also explains that a grammar is a type of pattern used "to recognize information in a document," whereas a structure is "an instantiation of a pattern in the document." A58(col.1:27-32). Thus, linking an action to a structure is not the same as associating an action with a grammar. *Bicon*, 441 F.3d at 950.

C. Claim 19 Should Not Have Been Rejected Because Nokia Does Not Disclose That The RPK's "Memory Contains Strings" And That The RPK Performs "Retrieving A String From The Memory And Scanning The Data To Identify The String"

There also is an independent reason to reverse the rejection of claim 19: Nokia does not disclose that the RPK's "memory contains strings" and that such strings are used in the step of "detecting a structure." Claim 19 claims: "The method recited in claim 15, wherein the memory contains strings, and wherein the step of detecting a structure further comprises the steps of retrieving a string from the memory and scanning the data to identify the string." A61(col.8:47-50). Nokia does not disclose this "string" method of detecting a structure.

The '647 patent specification teaches how strings can be used to detect a structure in a document. According to the specification, the computer stores a list

of strings—e.g., a list of "important names." A60(col.5:14). According to one embodiment, the strings can be stored in a "string library" in the memory. A60(col.5:14, col.6:44). When the computer receives the computer data to be scanned, the computer retrieves the strings from memory and searches for "strings in the data identical to those in the string library." A60(col.6:44-46); *see* A60(col.5:27). As claim 19 recites, the computer "receiv[es] computer data," "retriev[es] a string from the memory," and "scan[s] the data to identify the string." A61(col.8:27, col.8:49-50).

Thus, for example, the computer may have stored in memory a list of names or telephone numbers that are to be detected in a document. When a document (e.g., an e-mail) is received, the computer would search the document to see whether any of the stored names or telephone numbers is contained within the document. *See* A60(col.6:44-46).

That is different from how a grammar is used. According to the "grammar" method recited in claim 17, the computer detects structures in the data by "parsing the data based on the grammar." A61(col.8:40). Under the "string" method in claim 19, by contrast, the computer simply "scan[s] the data to identify the string" in the data. A61(col.8:50).

There is no evidence in Nokia that the RPK uses the string method to detect structures in data. The Board wrongly concluded that Nokia discloses the claimed

method simply because "Nokia discloses searching a message for a telephone number." A10. But the bare fact that Nokia searches for numbers in a message that might be a telephone number does not mean that it uses strings to do so. Nokia states that "the RPK searches for a point which could be a telephone number." A1260(col.6:22-23). It does so by looking for a numeral "followed by at least four of any of the characters included in the following set: numerals, -, /, (,), space," at least "three of the[] first five" of which "must be numerals." A1260(col.6:18-21). That use of a pattern or algorithm to parse the message in search of a potential telephone number is not the same as searching the message for a string that is identical to a string retrieved from memory, as in the '647 patent. A60(col.6:44-46).

The Board also concluded that Nokia discloses "a memory containing strings" because "Nokia discloses that the apparatus 'stores the number under a name in the directory' (col.8, ll.47-48)." A10. To be sure, Nokia discloses that *after* a telephone number is detected in the message and the user accepts the number, the number can be stored in the directory. A1260-A1261(col.5:32-43, col.8:27-31, col.8:47-54). But what claim 19 requires is that a string be used in "the step of detecting a structure" in the message in the first place. A61(col.8:48). Simply storing the detected structure in memory after it already is detected is not enough.

Finally, the Board erred in reasoning that Nokia discloses "a memory containing strings" simply because "in order to display and search the message in Nokia, the *message* would be 'stored' on the apparatus on which the search is performed." A10 (emphasis added). The fact that in Nokia the received text message would be stored in memory does not mean that Nokia uses strings. The Board appears to have conflated the text message and a string. But contrary to the Board's reasoning, the received data (i.e., the text message in Nokia) cannot be the same as the string because the string of the '647 patent is used to "detect[] a structure" in the received data. A61(col.8:48). The Board's rationale thus cannot withstand scrutiny.

CONCLUSION

For the foregoing reasons, the PTO's rejection of claims 13 and 15-31 of the '647 patent should be reversed.

Respectfully submitted,

JANUARY 31, 2014

/s/ Deanne E. Maynard

DEANNE E. MAYNARD
MARC A. HEARRON
MORRISON & FOERSTER LLP
2000 Pennsylvania Avenue, N.W.
Washington, D.C. 20006
Telephone: 202.887.8740

PETER J. YIM
MORRISON & FOERSTER LLP
425 Market Street
San Francisco, California 94105
Telephone: 415.268.7000

Counsel for Appellant Apple Inc.

ADDENDUM

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
90/011,287	10/15/2010	5946647	031486.647	9354

77970 7590 06/20/2013

Apple Inc.
c/o Novak Druce Connolly Bove + Quigg LLP
1000 Louisiana Street
Fifty-Third Floor
Houston, TX 77002

EXAMINER	
STEELMAN, MARY J	
ART UNIT	PAPER NUMBER
3992	

MAIL DATE	DELIVERY MODE
06/20/2013	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte APPLE, INC.,
Appellant and Patent Owner

Appeal 2013-001460
Reexamination Control 90/011,287
Patent 5,946,647
Technology Center 3900

Before MAHSHID D. SAADAT, KEVIN F. TURNER, and STEPHEN C. SIU, *Administrative Patent Judges*.

SIU, *Administrative Patent Judge*

DECISION ON APPEAL

Patent owner appeals under 35 U.S.C. §§ 134(b) and 306 from a final rejection of claims 13 and 15-31. We have jurisdiction under 35 U.S.C. §§ 134(b) and 306.

Appeal 2013-001460
Reexamination Control 90/011,287
Patent 5,946,647

STATEMENT OF THE CASE

This proceeding arose from a request for *ex parte* reexamination filed on October 15, 2010 of United States Patent 5,946,647 (the '647 Patent) issued to James R. Miller, Thomas Bonura, Bonnie Nardi, and David Wright on August 31, 1999. Presently, claims 13 and 15-31 stand rejected.

Patentee's invention relates to performing actions on structures in computer data (col. 1, ll. 10-11). Claim 13 reads as follows:

> 13. A program storage medium storing a computer program for causing a computer to perform the steps of:
>
> receiving computer data;
>
> detecting a structure in the data;
>
> linking at least one action to the detected structure;
>
> enabling selection of the structure and a linked action; and
>
> executing the selected action linked to the selected structure.

(App. Br. 23, Claims Appendix).

The Examiner cites the following references:

Nokia Mobile Phones Ltd. ("Nokia")	EP0458563 A2	Nov. 27, 1991

Pensoft Corporation, "Perspective Handbook," November 1992, ("Pensoft").

Appellant appeals the following rejections:

Claims 13, 15, 16, 20-22, and 24-31 stand rejected under 35 U.S.C.

§ 102(b) as anticipated by Pensoft (App. Br. 6).

Claims 13, 15, 17-23, and 25-31 stand rejected under 35 U.S.C. § 102(b) as anticipated by Nokia (App. Br. 10).

ISSUE

Did the Examiner err in rejecting claims 13 and 15-31?

PRINCIPLE OF LAW

In rejecting claims under 35 U.S.C. § 102, "[a] single prior art reference that discloses, either expressly or inherently, each limitation of a claim invalidates that claim by anticipation." *Perricone v. Medicis Pharm. Corp.,* 432 F.3d 1368, 1375 (Fed. Cir. 2005) (citation omitted).

ANALYSIS

Pensoft – Claims 13, 15, 16, 20-22, and 24-31

Appellant argues that "Pensoft does not disclose 'linking at least one action to the detected structure' as recited in claim 15" because, according to Appellant, "opening of a profile is not an action" and "there is no action being performed on the detected structure" (App. Br. 6 and 9).

The Examiner states that Pensoft discloses an "action" (i.e., opening a profile – Pensoft, pp. 9-12 and 43) that is "linked" to an item in a document (or a "structure" – see Pensoft, pp. 9-10) (Ans. 8-11). Appellant states that an "action" as recited in claim 15 is "a computer subroutine that causes the

CPU to perform a sequence of operations on the particular structure to which it is linked" (App. Br. 6, citing Specification, col. 2, ll. 32-34) and that Pensoft fails to disclose such an "action."

Even assuming Appellant's contention to be correct that an "action" must be a subroutine that causes the CPU to perform operations on a structure to which it is linked, Appellant has not persuasively demonstrated that the action of Pensoft (i.e., opening a profile) is not a subroutine that causes the CPU to perform operations. In fact, Pensoft discloses that upon selection of an item in a document, the computer (or CPU) opens a corresponding profile (*see e.g.*, Pensoft, pp. 9-12 and 43). In order for the computer to perform such an operation, the computer would have executed a program (or subroutine) to do so. Otherwise, the "action" of opening a profile would not be performed.

While Appellant argues that the profile of Pensoft "is not an action and does not contain an action" (App. Br. 7), Appellant does not indicate how opening the profile of Pensoft differs from an "action" (or an operation performed by a subroutine). One of ordinary skill in the art would have understood that "opening" a profile would have entailed an action at least because some activity would be performed to execute the operation.

Appellant also argues that "Pensoft does not disclose 'linking at least one action to the detected structure'" (App. Br. 9). As described above, in Pensoft, an item in a document is selected and a profile that corresponds to the selected item is launched. Appellant does not indicate a specialized

definition of the term "link" in the Specification. In the absence of such a definition, we construe the term broadly but reasonably and in light of the Specification to include "associated or connected with."

In Pensoft, the profile is "linked" to the selected item because the specific profile that corresponds to the selected item (and not an unrelated profile) is launched upon selection of the item. Hence, the opening of the specific profile is "linked to" (or associated with) the corresponding selected item.

While Appellant argues that "Pensoft uses 'link' to refer to the relationship between two pieces of information" (App. Br. 10), Appellant does not indicate how opening a profile in Pensoft is not "linked to" (or associated with) the specifically selected item that corresponds to the profile that is opened.

Appellant does not provide additional arguments in support of claims 13, 16, 20-22, and 24-31.

Affirmance of the rejection for claims 13, 15, 16, 20-22, and 24-31 based on Pensoft renders it unnecessary to reach the propriety of the Examiner's decision to reject those claims on a different basis. *Cf. In re Gleave*, 560 F.3d 1331, 1338 (Fed. Cir. 2009). As such, we need not reach the propriety of the rejection of those claims over Nokia.

Nokia- Claims 17-19 and 23

Appellant argues that Nokia fails to disclose linking (App. Br. 11-14). Claim 15 (from which claims 17-19 and 23 depend), for example, recites detecting a structure in computer data and linking an action to the detected structure. We disagree with Appellant for at least the reasons set forth by the Examiner (*see e.g.*, Ans. 17-21).

Appellant appears to argue that the Specification discloses that to "link" actions with structures means "to associate actions with . . . structures" (App. Br. 11, citing '647 patent, col. 3, ll. 38-44). Even assuming Appellant's contention to be true that "linking" as recited in claim 15 includes "associating," we agree with the Examiner that Nokia discloses this feature.

For example, Nokia discloses that an apparatus (i.e., "RPK") "makes a call to [a telephone] number" (col. 5, l. 37) after the apparatus "searches out" the telephone number in a message and "the user accepts the number" (col. 5, ll. 34-36). Nokia further discloses that after the user "accepts the number (col. 7, l. 15), the "RPK makes a call to the number" (col. 8, l. 1) "only when the number to be used is . . . in the display as a consequence of [the user accepting the number]" (col. 8, ll. 3-5). The number is associated with (or "linked") to the action of the user accepting the number because the apparatus makes a call to the number only "as a consequence of" the user performing the action (i.e., accepting the number).

Appellant also argues that Nokia fails to disclose enabling selection of the structure and a linked action, as recited in claim 15 (from which claims 17-19 and 23 depend) (App. Br. 15). We disagree with Appellant for at least the reasons set forth by the Examiner (*see e.g.*, Ans. 17-21).

As indicated by the Examiner and described above, Nokia discloses a user who "accepts the [telephone] number" after the apparatus "searches out" the telephone number in a message (col. 5, ll. 34-35) and the apparatus calling the telephone number "as a consequence" (col. 8, ll. 4-5) of the user accepting the telephone number. Appellant has not sufficiently demonstrated that the user in Nokia would have been able to accept a telephone number if the apparatus does not enable the user to select the telephone number (or "structure"). One of ordinary skill in the art would have understood that in order for a user to be able to "accept" a telephone number on an apparatus (and subsequent calling of the telephone number "as a consequence" of the user accepting the telephone number), the apparatus would enable the user to do so. Otherwise, the user would be unable to accept the telephone number on the apparatus, which would be contrary to the disclosure in Nokia.

Appellant also argues that Nokia fails to disclose executing the selected action linked to the selected structure, as recited in claim 15 (App. Br. 16). We disagree with Appellant for at least the reasons set forth by the Examiner (*see e.g.*, Ans. 23).

As indicated by the Examiner and as described above, Nokia discloses a user accepting a telephone number on an apparatus and the apparatus calling the telephone number "as a consequence" of the user accepting the telephone number. Appellant does not indicate a difference between a user accepting a telephone number for subsequent calling of the selected telephone number and "executing" a selected action. In both cases, an action is selected and the action is performed or "executed."

Regarding claim 18, Appellant argues that Nokia fails to disclose "that 'the grammar is associated with a particular action, and wherein the step of linking at least one action to the detected structure includes the step of linking the particular action to the detected structure'" (App. Br. 17). We disagree with Appellant for at least the reasons set forth by the Examiner (*see e.g.*, Ans. 25-28).

Appellant argues that claim 18 "requires a grammar associated with a particular action" (App. Br. 17), thus implying that Nokia fails to disclose a grammar associated with an action. However, as described above and as indicated by the Examiner, Nokia discloses a telephone number (or a "grammar") that is "associated with" an action (e.g., selecting the telephone number and calling the telephone number as a consequence of the selecting of the telephone number). Appellant does not demonstrate a difference between the telephone number (or structure of the telephone number) and the "grammar" as recited in claim 18.

Regarding claim 19, Appellant argues that Nokia fails to disclose "that 'the memory contains strings, and wherein the step of detecting a structure further comprises the steps of retrieving a string from the memory and scanning the data to identify the string'" (App. Br. 18). We disagree with Appellant for at least the reasons set forth by the Examiner (*see e.g.,* Ans. 30).

Appellant argues that "Nokia looks for . . . four sequential number" in a text message but fails to disclose "use of a string to identify that exact string" (App. Br. 18). However, as the Examiner indicates and as described above, Nokia discloses searching a message for a telephone number. Appellant has not sufficiently pointed out any differences between searching a message for a telephone number and scanning data (or a message) to identify a string (or telephone number).

Appellant also argues that "Nokia fails to disclose a memory containing strings" (App. Br. 18). However, as the Examiner indicates and as described above, Nokia discloses that the apparatus "stores the number under a name in the directory" (col. 8, ll. 47-48). Also, Appellant does not indicate how the message that is searched in Nokia is not stored in the apparatus. One of skill in the art would have understood that in order to display and search the message in Nokia, the message would be "stored" on the apparatus on which the search is performed. Otherwise, the apparatus would be unable to search the message (the message not being stored and therefore not being accessible).

Appeal 2013-001460
Reexamination Control 90/011,287
Patent 5,946,647

Appellant does not provide additional arguments in support of claims 17-19 and 23.

CONCLUSION

We conclude that the Examiner did not err in rejecting claims 13 and 15-31.

DECISION

The decision of the Examiner to reject claims 13, 15, 16, 20-22, and 24-31 as anticipated by Pensoft and claims 17-19 and 23 as anticipated by Nokia is AFFIRMED.

Requests for extensions of time in this *ex parte* reexamination proceeding are governed by 37 C.F.R. § 1.550(c). *See* 37 C.F.R. § 41.50(f).

AFFIRMED

peb

US005946647A

United States Patent [19]

Miller et al.

[11] **Patent Number: 5,946,647**

[45] **Date of Patent: Aug. 31, 1999**

[54] **SYSTEM AND METHOD FOR PERFORMING AN ACTION ON A STRUCTURE IN COMPUTER-GENERATED DATA**

[75] Inventors: **James R. Miller**, Mountain View; **Thomas Bonura**, Capitola; **Bonnie Nardi**, Mountain View; **David Wright**, Santa Clara, all of Calif.

[73] Assignee: **Apple Computer, Inc.**, Cupertino, Calif.

[21] Appl. No.: **08/595,257**

[22] Filed: **Feb. 1, 1996**

[51] **Int. Cl.**[6] **G06F 17/27**

[52] **U.S. Cl.** **704/9**; 704/1

[58] **Field of Search** 704/1, 7, 9–10, 704/243; 707/513, 101–104

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,115,390	5/1992	Fukuda et al.	364/146
5,130,924	7/1992	Barker et al.	704/1
5,164,899	11/1992	Sobotka et al.	704/9
5,202,828	4/1993	Vertelney et al.	364/419
5,247,437	9/1993	Vale et al.	704/1
5,369,575	11/1994	Lamberti et al.	704/1
5,574,843	11/1996	Gerlach et al.	395/118

OTHER PUBLICATIONS

TerryMorse Software "What is Myrmidon" Downloaded from the Internet at URL http://www.terrymorse.com (Publication Date Unknown), 2 pages.

Shoens, K. et al. "Rufus System: Information Organization for Semi–Structured Data," Proceedings of the 19th VLDB Conference (Dublin, Ireland 1993), pp. 1–12.

Schwarz, Peter and Shoens, Kurt. "Managing Change in the Rufus System," Abstract from the IBM Almaden Research Center, pp. 1–16.

Myers, Brad A. "Tourmaline: Text Formatting by Demonstration," (Chapter 14) in *Watch What I Do: Programming by Demonstration,* edited by Allen Cypher, MIT Press, (Cambridge, MA 1993), pp. 309–321.

Maulsby, David. "Instructible Agents," Dissertation from the Department of Computer Science at The University of Calgary (Calgary, Alberta—Jun. 1994), pp. 178, 181–188, 193–196 (from Chapter 5).

Rus, Daniela and Subramanian, Devika. "Designing Structure–Based Information Agents," AAAI Symposium (Mar. 1994), pp. 79–86.

Primary Examiner—Forester W. Isen

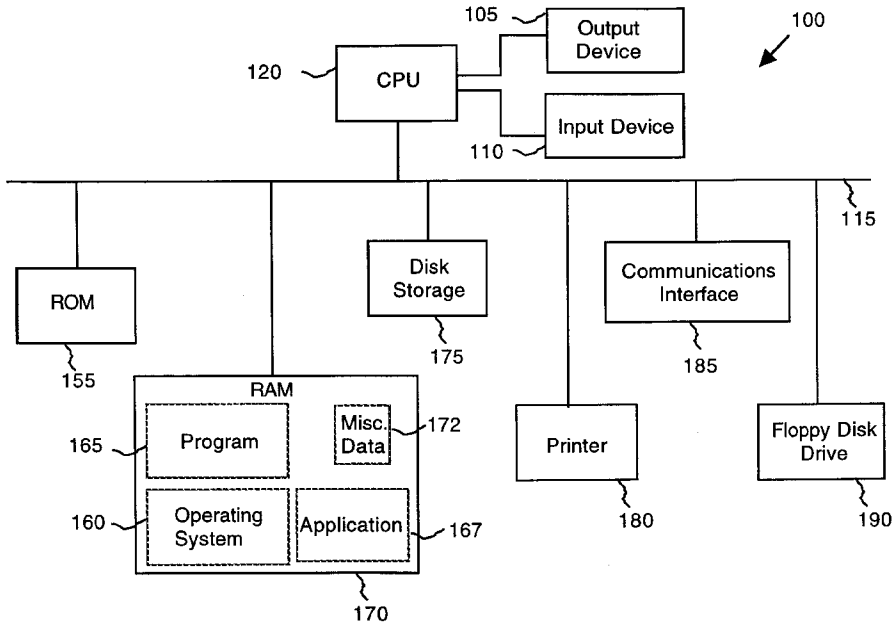
Assistant Examiner—Patrick N. Edouard

Attorney, Agent, or Firm—Carr & Ferrell LLP

[57] **ABSTRACT**

A system and method causes a computer to detect and perform actions on structures identified in computer data. The system provides an analyzer server, an application program interface, a user interface and an action processor. The analyzer server receives from an application running concurrently data having recognizable structures, uses a pattern analysis unit, such as a parser or fast string search function, to detect structures in the data, and links relevant actions to the detected structures. The application program interface communicates with the application running concurrently, and transmits relevant information to the user interface. Thus, the user interface can present and enable selection of the detected structures, and upon selection of a detected structure, present the linked candidate actions. Upon selection of an action, the action processor performs the action on the detected structure.

24 Claims, 10 Drawing Sheets



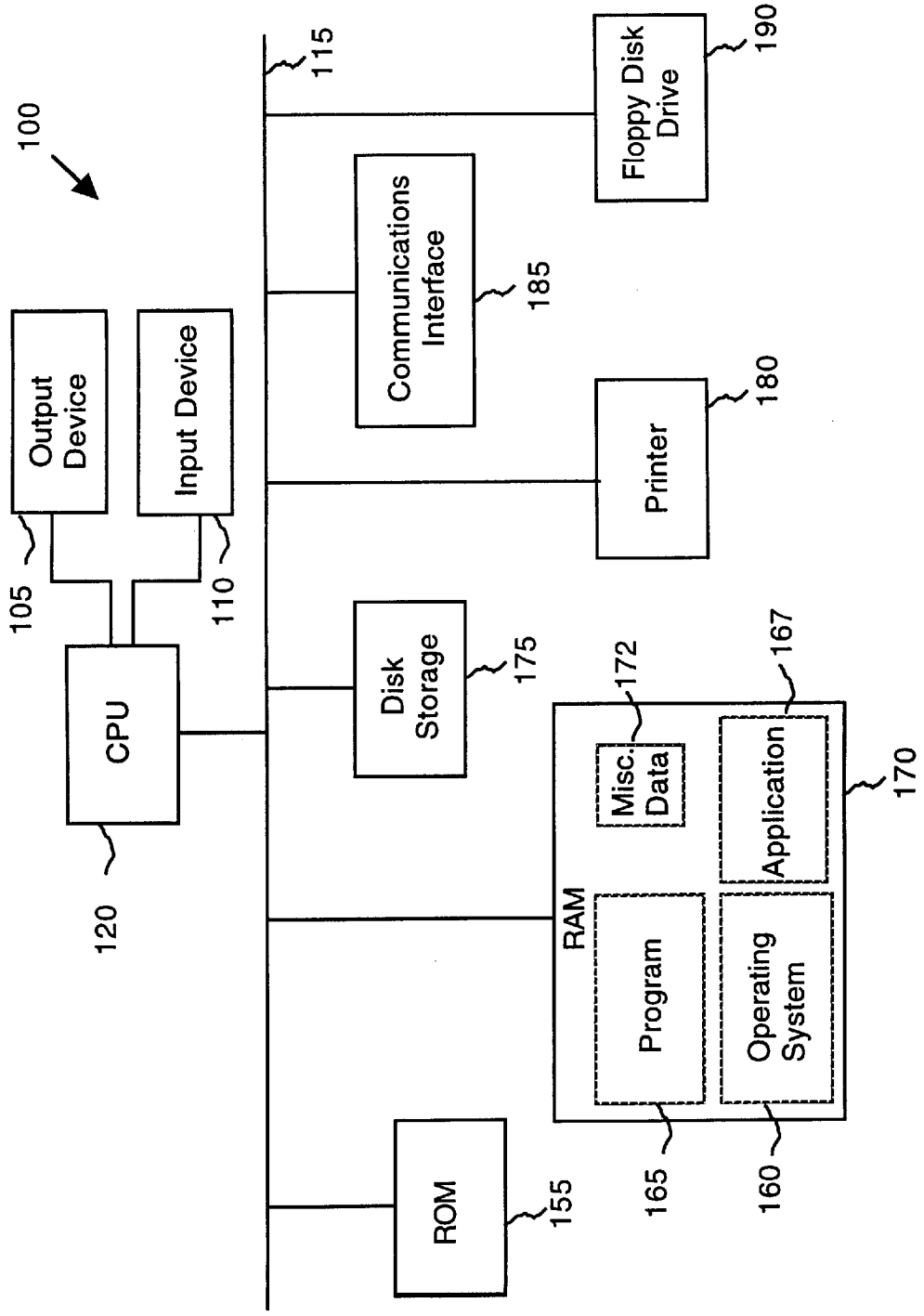

100
115
Floppy Disk Drive
190
Communications Interface
185
Output Device
Input Device
Printer
180
105
110
CPU
120
Disk Storage
175
172
167
Misc. Data
Application
170
RAM
Program
Operating System
165
160
ROM
155

FIG. 1

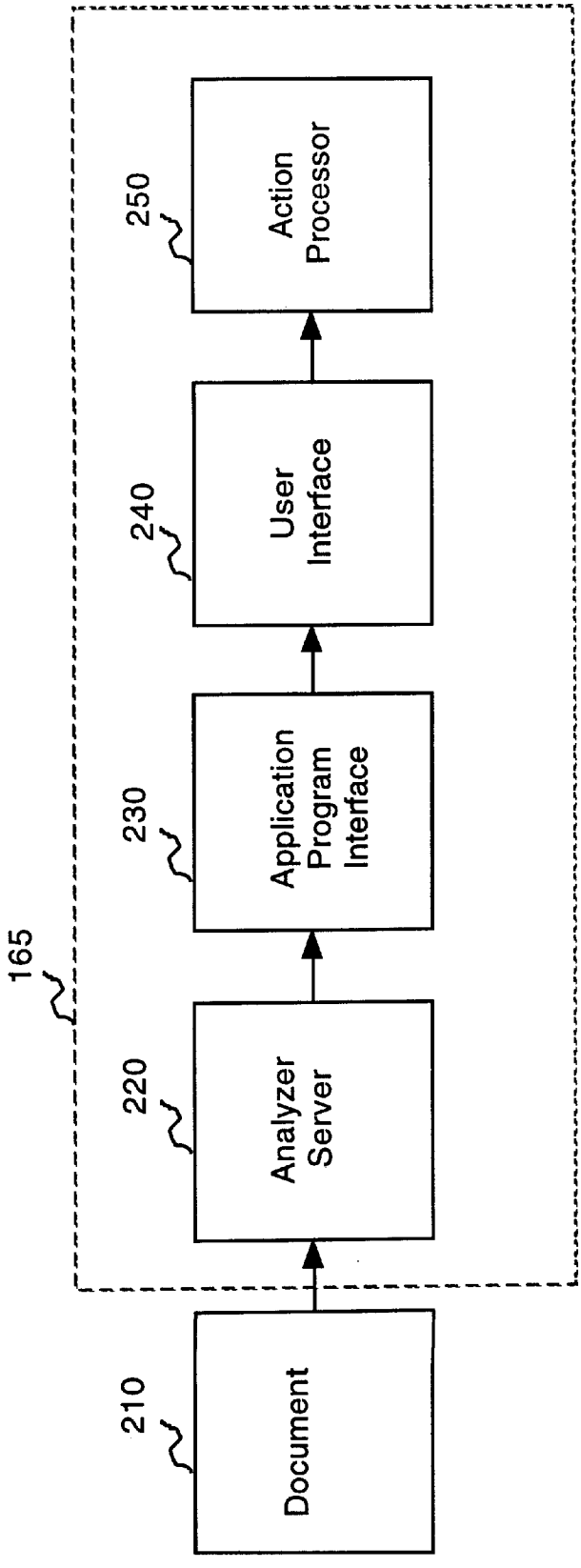

165
210
Document
220
Analyzer Server
230
Application Program Interface
240
User Interface
250
Action Processor

FIG. 2

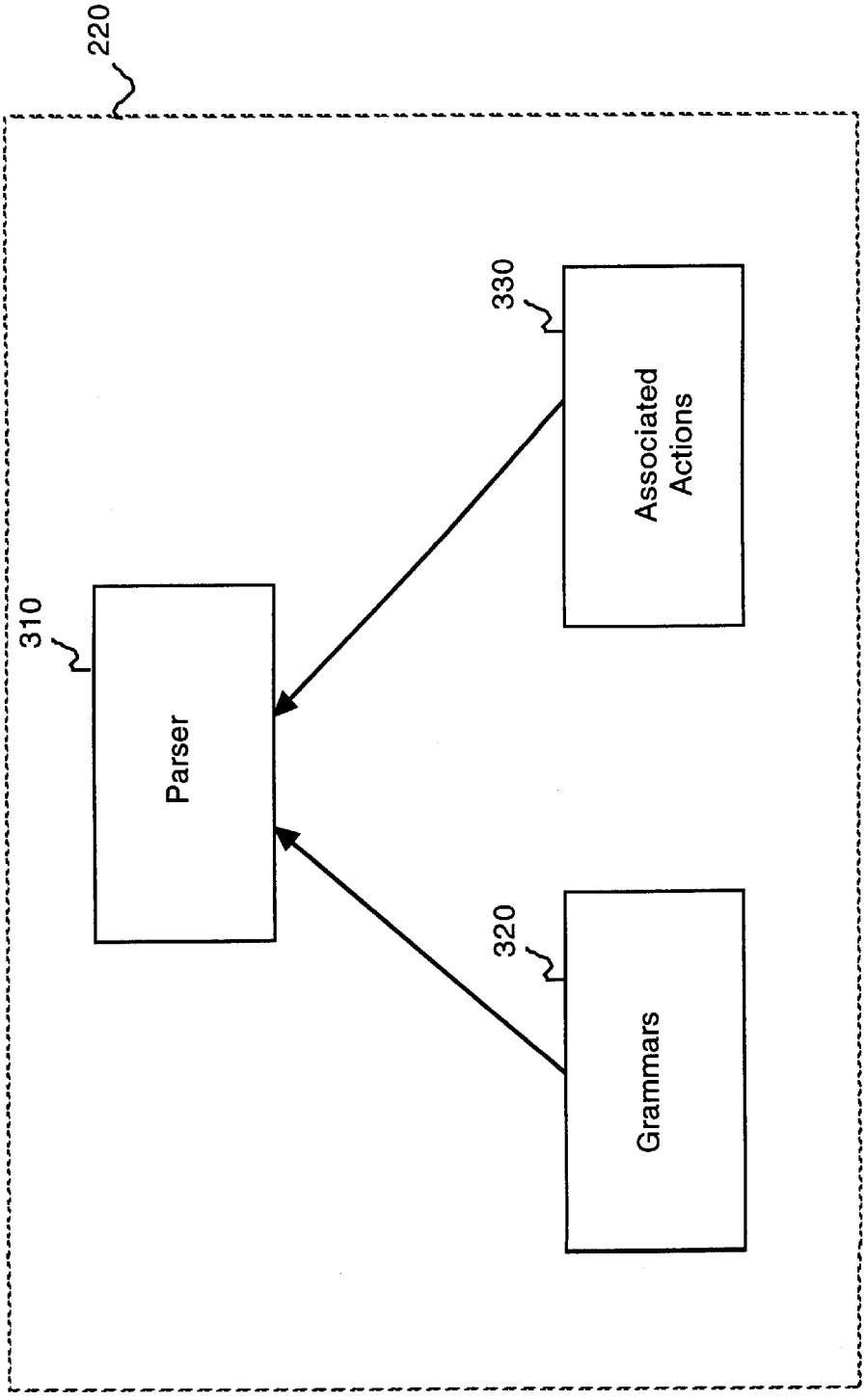


FIG. 3

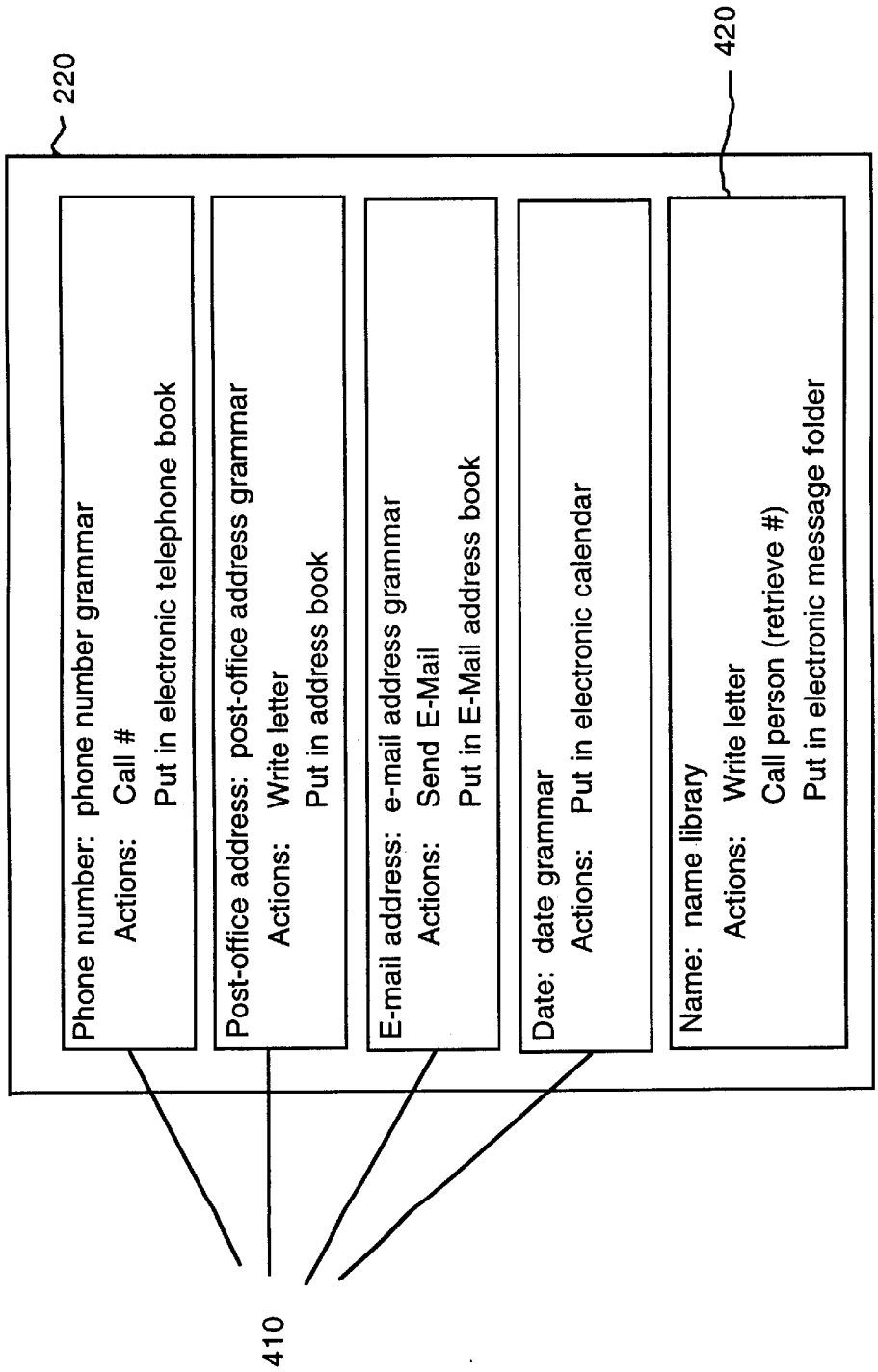


FIG. 4

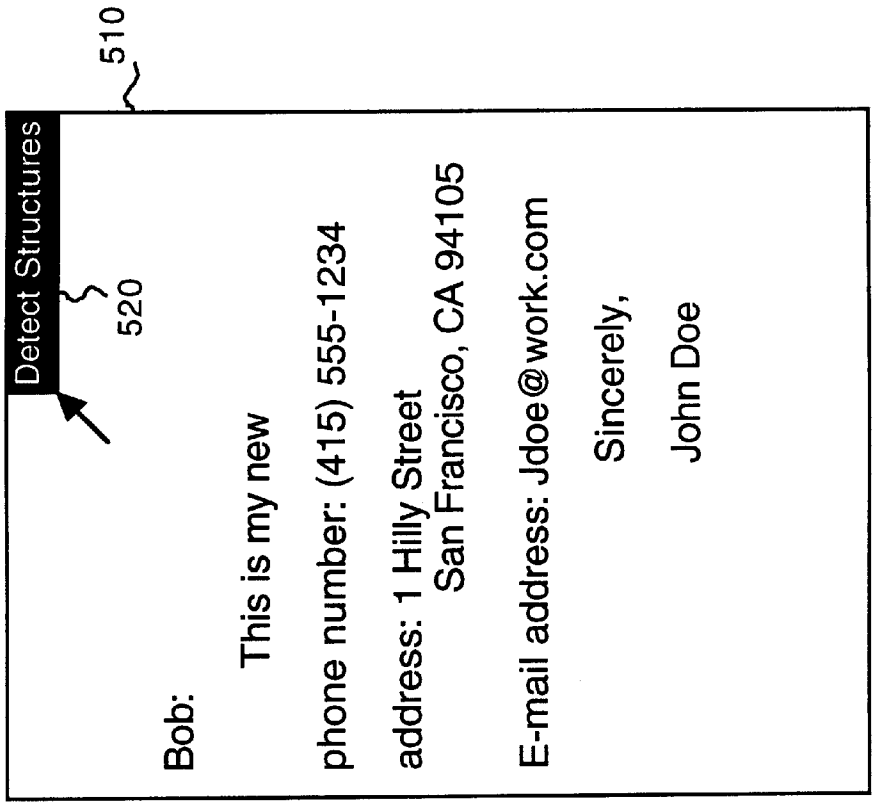

510
Detect Structures
520
Bob:
This is my new
phone number: (415) 555-1234
address: 1 Hilly Street
San Francisco, CA 94105
E-mail address: Jdoe@work.com
Sincerely,
John Doe

FIG. 5

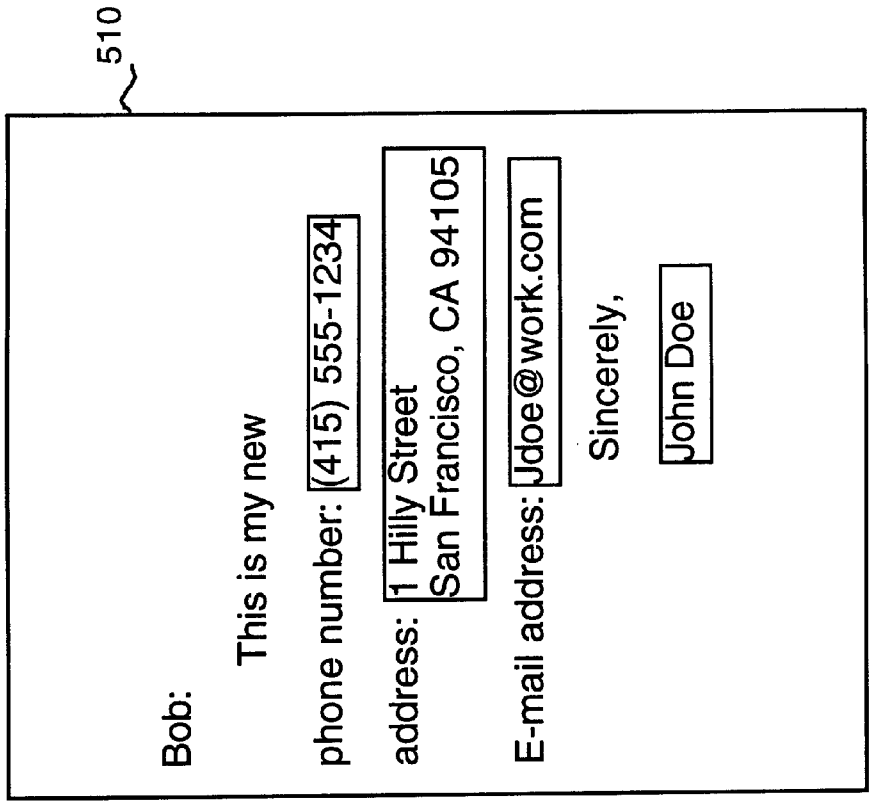

510
Bob:
This is my new
phone number: (415) 555-1234
address: 1 Hilly Street
San Francisco, CA 94105
E-mail address: Jdoe@work.com
Sincerely,
John Doe

FIG. 6

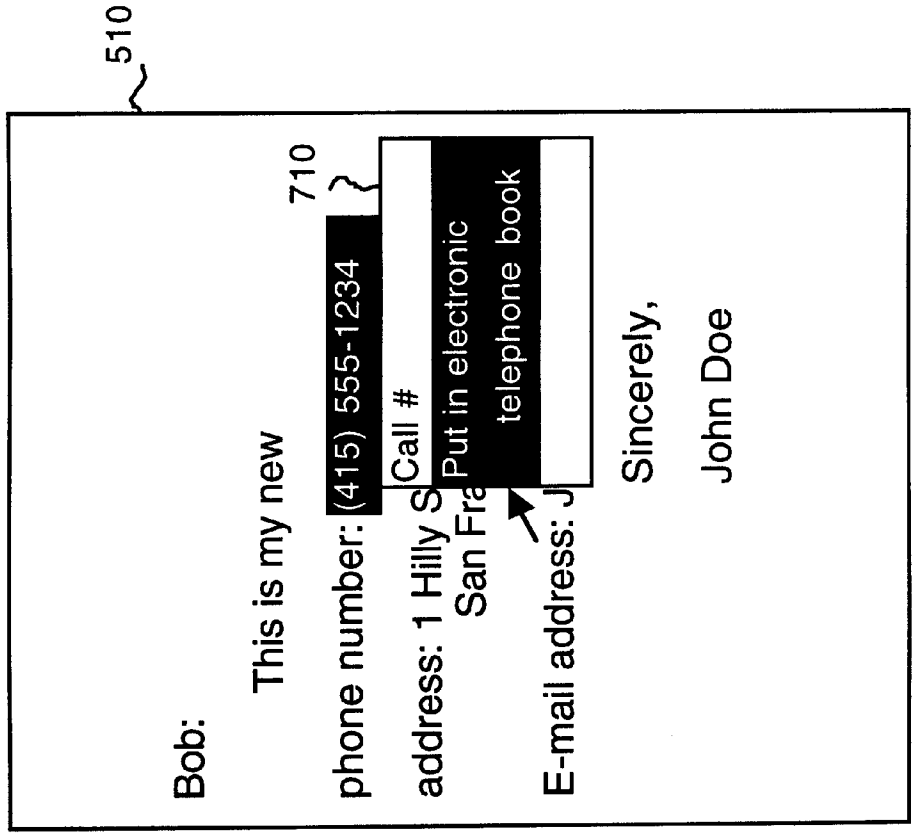

510
710
Bob:
This is my new
phone number: (415) 555-1234
address: 1 Hilly S
San Fra
E-mail address: J
Call #
Put in electronic
telephone book
Sincerely,
John Doe

FIG. 7

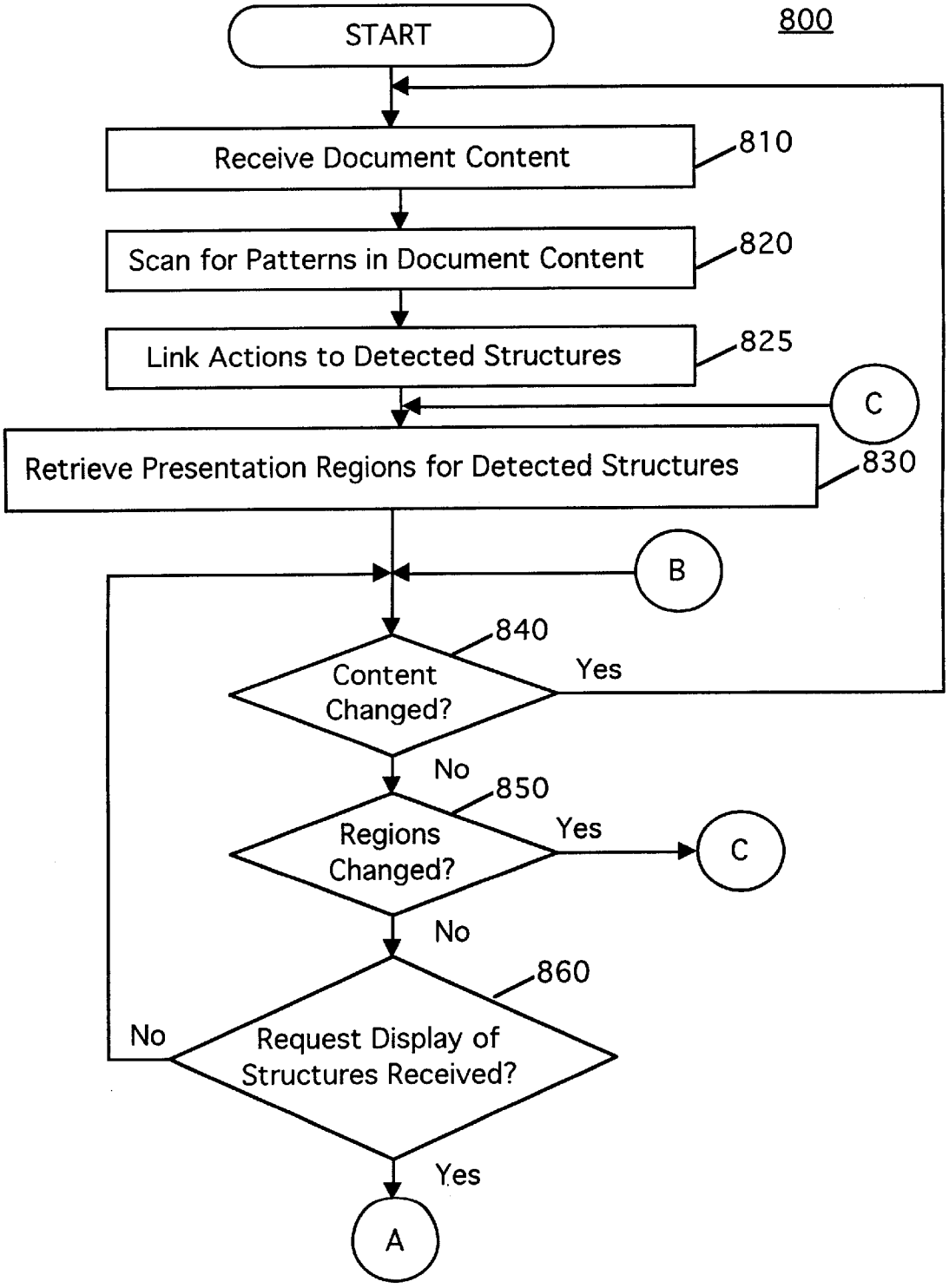

800
START
Receive Document Content
810
Scan for Patterns in Document Content
820
Link Actions to Detected Structures
825
C
Retrieve Presentation Regions for Detected Structures
830
B
840
Content Changed?
Yes
No
850
Regions Changed?
Yes
C
No
860
No
Request Display of Structures Received?
Yes
A

FIG. 8

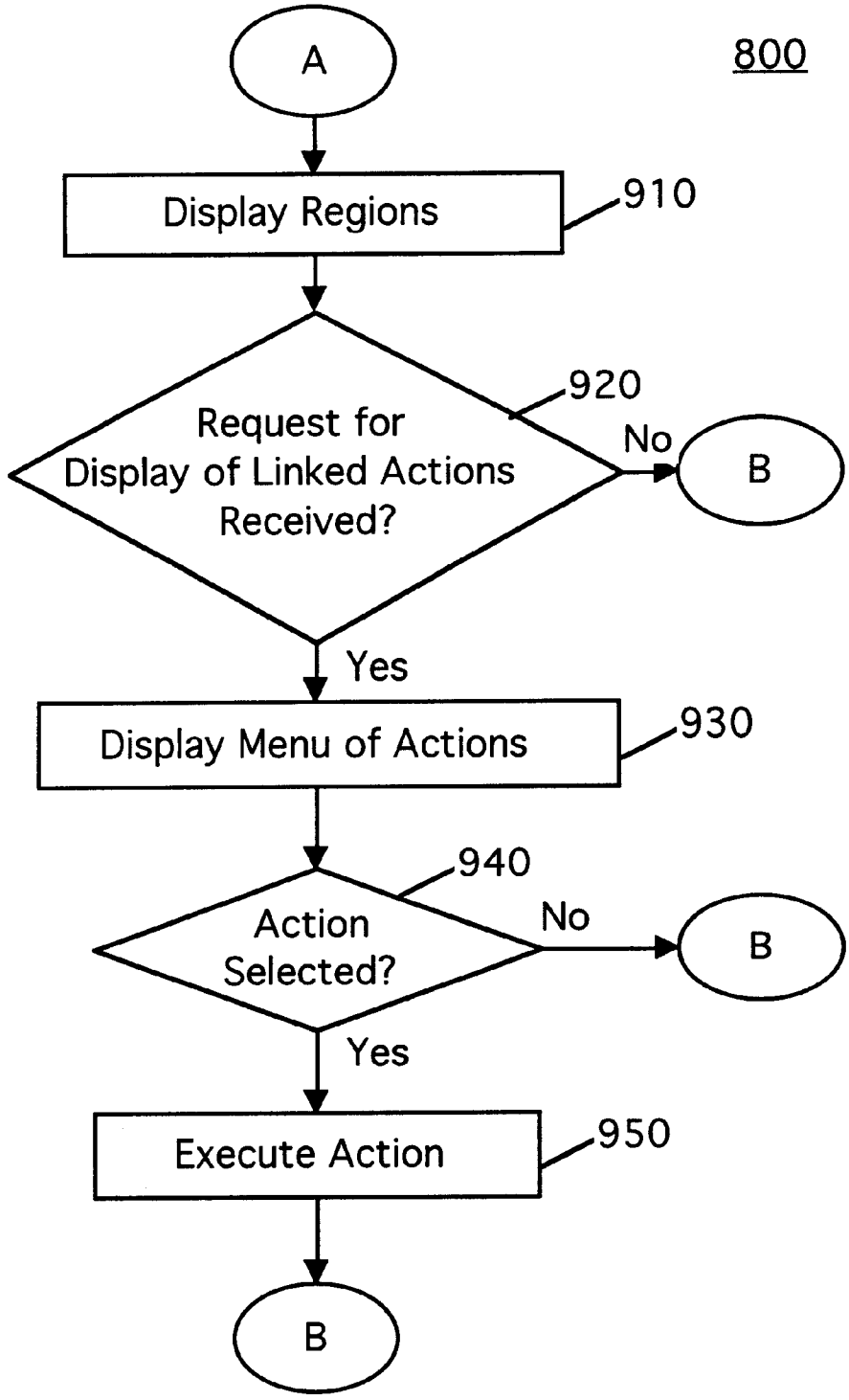

A
800
Display Regions
910
Request for Display of Linked Actions Received?
920
No
B
Yes
Display Menu of Actions
930
Action Selected?
940
No
B
Yes
Execute Action
950
B

FIG. 9

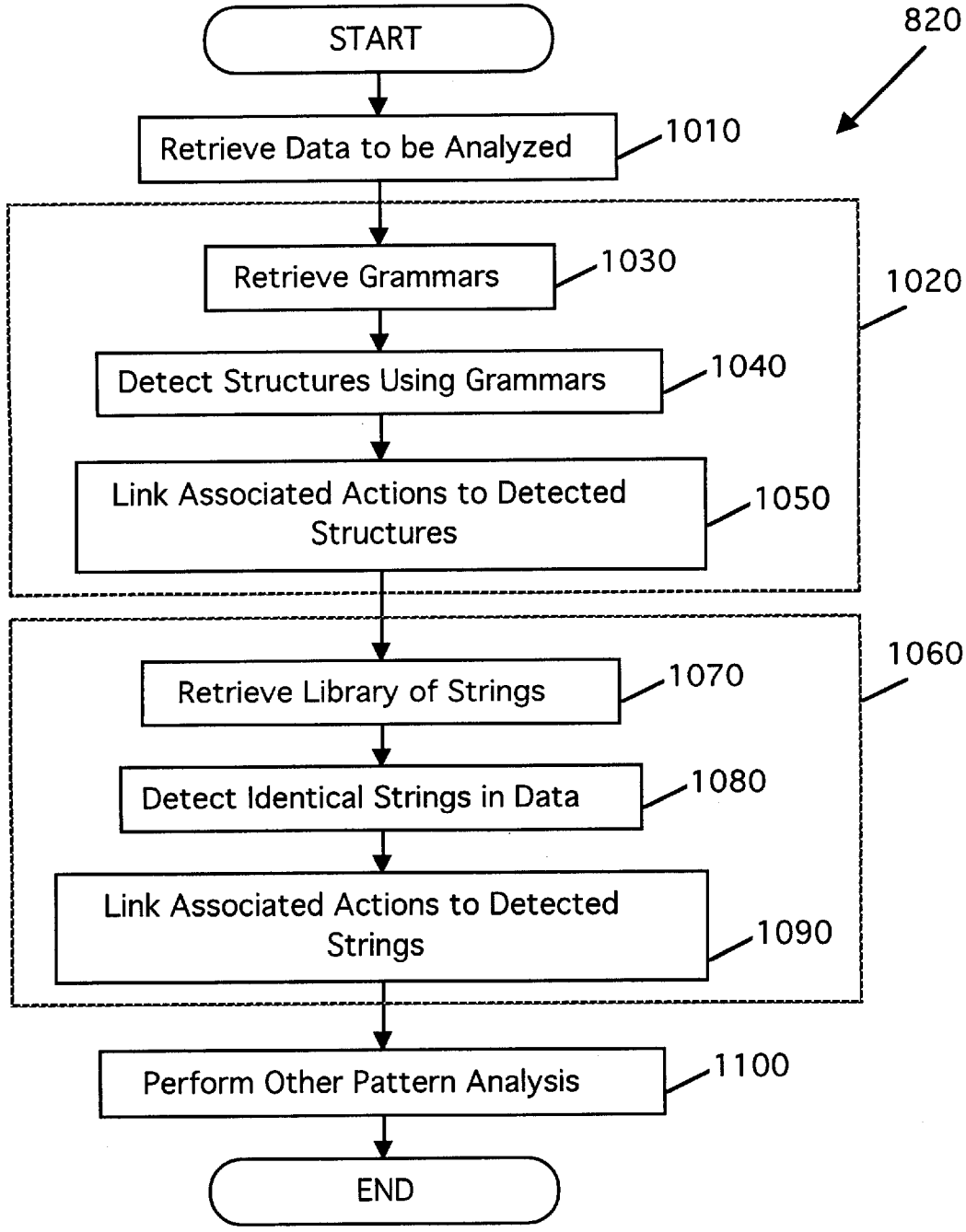

START
820
Retrieve Data to be Analyzed
1010
Retrieve Grammars
1030
1020
Detect Structures Using Grammars
1040
Link Associated Actions to Detected Structures
1050
1060
Retrieve Library of Strings
1070
Detect Identical Strings in Data
1080
Link Associated Actions to Detected Strings
1090
Perform Other Pattern Analysis
1100
END

FIG. 10

SYSTEM AND METHOD FOR PERFORMING AN ACTION ON A STRUCTURE IN COMPUTER-GENERATED DATA

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates generally to manipulation of structures in computer data. More particularly, the invention relates to a system and method for performing computer-based actions on structures identified in computer data.

2. Description of the Background Art

Much data that appears in a computer user's day-to-day activities contains recognizable structures that have semantic significance such as phone numbers, e-mail addresses, post-office addresses, zip codes and dates. In a typical day, for example, a user may receive extensive files from word-processing programs and e-mail that contain several of these structures. However, visually searching data files or documents to find these structures is laborious and cognitively disruptive, especially if the document is lengthy and hard to follow. Furthermore, missing a structure such as a date may lead to missing an important meeting or missing a deadline.

To help facilitate searching a document for these structures, programmers can create or employ pattern analysis units, such as parsers, to automatically identify the structures. For the purposes of the present description, the term "pattern" refers to data, such as a grammar, regular expression, string, etc., used by a pattern analysis unit to recognize information in a document, such as dates, addresses, phone numbers, names, etc. The term "structure" refers to an instantiation of a pattern in the document. That is, a "date" pattern will recognize the structure "Oct. 31, 1995." The application of a pattern to a document is termed "parsing."

Conventional systems that identify structures in computer data do not enable automatic performance of an action on an identified structure. For example, if a long e-mail message is sent to a user, the user may implement a pattern analysis unit to search for particular structures, such as telephone numbers. Upon identification of a structure, the user may want to perform an action on the structure, such as moving the number to an electronic telephone book. This usually involves cutting the structure from the e-mail message, locating and opening the electronic telephone book application program, pasting the structure into the appropriate field, and closing the application program. However, despite the fact that computer systems are getting faster and more efficient, this procedure is still tedious and cognitively disruptive.

One type of system that has addressed this problem involves detecting telephone numbers. Such systems enable a user to select a telephone number and request that the application automatically dial the number. However, these systems do not recognize the selected data as a telephone number, and they generally produce an error message if the user selects invalid characters as a phone number. Also, they do not enable the performance of other candidate actions, such as moving the number to an electronic telephone book. That is, if a user wishes to perform a different action on an identified telephone number, such as storing the number in an address book, the user cannot automatically perform the action but must select and transfer the number to the appropriate data base as described above.

Therefore, a system is needed that identifies structures, associates candidate actions to the structures, enables selection of an action and automatically performs the selected action on the structure.

SUMMARY OF THE INVENTION

The present invention overcomes the limitations and deficiencies of previous systems with a system that identifies structures in computer data, associates candidate actions with each detected structure, enables the selection of an action, and automatically performs the selected action on the identified structure. It will be appreciated that the system may operate on recognizable patterns for text, pictures, tables, graphs, voice, etc. So long as a pattern is recognizable, the system will operate on it. The present invention has significant advantages over previous systems, in that the present system may incorporate an open-ended number and type of recognizable patterns, an open-ended number and type of pattern analysis units, and further that the system may enable an open-ended number and type (i.e. scripts, macros, code fragments, etc.) of candidate actions to associate with, and thus perform, on each identified structure.

The present invention provides a computer system with a central processing unit (CPU), input/output (I/O) means, and a memory that includes a program to identify structures in a document and perform selected computer-based actions on the identified structures. The program includes program subroutines that include an analyzer server, an application program interface, a user interface and an action processor. The analyzer server receives data from a document having recognizable structures, and uses patterns to detect the structures. Upon detection of a structure, the analyzer server links actions to the detected structure. Each action is a computer subroutine that causes the CPU to perform a sequence of operations on the particular structure to which it is linked. An action may specify opening another application, loading the identified structure into an appropriate field, and closing the application. An action may further include internal actions, such as storing phone numbers in an electronic phone book, addresses in an electronic address book, appointments on an electronic calendar, and external actions such as returning phone calls, drafting letters, sending facsimile copies and e-mail, and the like.

Since the program may be executed during the run-time of another program, i.e. the application which presents the document, such as Microsoft Word, an application program interface provides mechanisms for interprogram communications. The application program interface retrieves and transmits relevant information from the other program to the user interface for identifying, presenting and enabling selection of detected structures. Upon selection of a detected structure, the user interface presents and enables selection of candidate actions. When a candidate action is selected, the action processor performs the selected action on the selected structure.

In addition to the computer system, the present invention also provides methods for performing actions on identified structures in a document. In this method, the document is analyzed using a pattern to identify corresponding structures. Identified structures are stored in memory and presented to the user for selection. Upon selection of an identified structure, a menu of candidate actions is presented, each of which may be selected and performed on the selected structure.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram of a computer system having a program stored in RAM, in accordance with the present invention.

FIG. **2** is a block diagram of the program of FIG. **1**.

FIG. **3** is a block diagram illustrating the analyzer server of FIG. **2**.

FIG. **4** is a block diagram illustrating a particular example of the analyzer server of FIG. **2**.

FIG. **5** illustrates a window presenting an example of a document having recognizable structures.

FIG. **6** illustrates a window with the identified structures in the example document of FIG. **5** highlighted based on the analyzer server of FIG. **4**.

FIG. **7** illustrates a window showing the display of a pop-up menu for selecting an action.

FIGS. **8** and **9** together are a flowchart depicting the preferred method for selecting and performing an action on an identified structure.

FIG. **10** is a flowchart depicting the preferred method for identifying a structure in a data sample.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Referring now to FIG. **1**, a block diagram is shown of a computer system **100** including a CPU **120**. Computer system **100** is preferably a microprocessor-based computer, such as a Power Macintosh manufactured by Apple Computer, Inc. of Cupertino, Calif. An input device **110**, such as a keyboard and mouse, and an output device **105**, such as a CRT or voice module, are coupled to CPU **120**. ROM **155**, RAM **170** and disk storage **175** are coupled to CPU **120** via signal bus **115**. Computer system **100** optionally further comprises a printer **180**, a communications interface **185**, and a floppy disk drive **190**, each coupled to CPU **120** via signal bus **115**.

Operating system **160** is a program that controls and facilitates the processing carried out by CPU **120**, and is typically stored in RAM **170**. Application **167** is a program, such as a word-processor or e-mail program, that presents data on output device **105** to a user. The program **165** of the present invention is stored in RAM **170** and causes CPU **120** to identify structures in the data presented by application **167**, to associate actions with the structures identified in the data, to enable the user to select a structure and an action, and to automatically perform the selected action on the identified structure. This program **165** may be stored in disk storage **175** and loaded into an allocated section of RAM **170** prior to execution by CPU **120**. Another section of RAM **170** is used for storing intermediate results and miscellaneous data **172**. Floppy disk drive **190** enables the storage of the present program **165** onto a removable storage medium which may be used to initially load program **165** into computer system **100**.

Referring now to FIG. **2**, a schematic block diagram of program **165** is shown together with its interaction with a document **210**. Program **165** contains program subroutines including an analyzer server **220**, an application program interface **230**, a user interface **240** and an action processor **250**. Analyzer server **220** receives data having recognizable patterns from a document **210**, which may be retrieved from a storage medium such as RAM **170**, ROM **155**, disk storage **175**, or the like, and presented on output device **105** by application **167**. Analyzer server **220** comprises one or more pattern analysis units, such as a parser and grammars or a fast string search function and dictionaries, which uses patterns to parse document **210** for recognizable structures. Upon detection of a structure, analyzer server **220** links actions associated with the responsible pattern to the detected structure, using conventional pointers.

After identifying structures and linking actions, application program interface **230** communicates with application **167** to obtain information on the identified structures so that user interface **240** can successfully present and enable selection of the actions. In a display-type environment, application program interface **230** retrieves the locations in document **210** of the presentation regions for the detected structures from application **167**. Application program interface **230** then transmits this location information to user interface **240**, which highlights the detected structures, although other presentation mechanisms can be used. User interface **240** enables selection of an identified structure by making the presentation regions mouse-sensitive, i.e. aware when a mouse event such as a mouse-down operation is performed while the cursor is over the region. Alternative selection mechanisms can be used such as touch sensitive screens and dialog boxes. It will be appreciated that detected structures can be hierarchical, i.e. that a sub-structure can itself be selected and have actions associated with it. For example, a user may be able to select the year portion of an identified date, and select actions specific to the year rather than to the entire date.

User interface **240** communicates with application **167** through application program interface **230** to determine if a user has performed a mouse-down operation in a particular mouse-sensitive presentation region, thereby selecting the structure presented at those coordinates. Upon selection of this structure, user interface **240** presents and enables selection of the linked candidate actions using any selection mechanism, such as a conventional pull-down or pop-up menu.

The above description of the user interface is cast in terms of a purely visual environment. However, the invention is not limited to visual interface means. For example, in an audio environment, user interface **240** may present the structures and associated actions to the user using voice synthesis and may enable selection of a pattern and action using voice or sound activation. In this type of embodiment, analyzer server **220** may be used in conjunction with a text-to-speech synthesis application **167** that reads documents to users over a telephone. Analyzer server **220** scans document **210** to recognize patterns and link actions to the recognized patterns in the same manner as described above. In the audio environment, user interface **240** may provide a special sound after application **167** reads a recognized pattern, and enable selection of the pattern through the use of an audio interface action, such as a voice command or the pressing of a button on the touch-tone telephone keypad as before. Thus, user interface **240** may present the linked actions via voice synthesis. One can create various environments having a combination of sensory mechanisms.

Upon selection of a candidate action, user interface **240** transmits the selected structure and the selected action to action processor **250**. Action processor **250** retrieves the sequence of operations that constitute the selected action, and performs the sequence using the selected structure as the object of the selected action.

Referring now to FIG. **3**, a block diagram illustrating an analyzer server **220** is shown. In this figure, analyzer server **220** is described as having a parser **310** and a grammar file **320**, although alternatively or additionally a fast string search function or other function can be used. Parser **310** retrieves a grammar from grammar file **320** and parses text using the retrieved grammar. Upon identification of a structure in the text, parser **310** links the actions associated with the grammar to the identified structure. More particularly, parser **310** retrieves from grammar file **320** pointers attached

to the grammar and attaches the same pointers to the identified structure. These pointers direct the system to the associated actions contained in associated actions file **330**. Thus, upon selection of the identified structure, user interface **240** can locate the linked actions.

FIG. **4** illustrates an example of an analyzer server **220**, which includes grammars **410** and a string library **420** such as a dictionary, each with associated actions. One of the grammars **410** is a telephone number grammar with associated actions for dialing a number identified by the telephone number grammar or placing the number in an electronic telephone book. Analyzer server **220** also includes grammars for post-office addresses, e-mail addresses and dates, and a string library **420** containing important names. When analyzer server **220** identifies an address using the "e-mail address" grammar, actions for sending e-mail to the identified address and putting the identified address in an e-mail address book are linked to the address.

FIG. **5** shows a window **510** presenting an exemplary document **210** having data containing recognizable structures, including a phone number, post-office address, e-mail address, and name. Window **510** includes a button **520** for initiating program **165**, although alternative mechanisms such as depressing the "option" key may be used. Upon initiation of program **165**, system **100** transmits the contents of document **210** to analyzer server **220**, which parses the contents based on grammars **410** and strings **420** (FIG. **4**). This parsing process produces the window shown in FIG. **6**. As illustrated in FIG. **6**, analyzer server **220** identifies the phone number, post-office address, e-mail address and name. Although not shown in FIG. **6**, analyzer server **220** links the actions associated with grammars **410** and strings **420** to these identified structures, and application program interface **230** retrieves information on the location of these structures from application **167**. User interface **240** then highlights the identified structures in document **210**, and makes the identified structures mouse-sensitive.

As shown in FIG. **7**, upon recognition of a mouse-down operation over a structure, user interface **240** presents a pop-up menu **710**. In this example, pop-up menu **710** displays the candidate actions linked to the selected telephone number grammar **410**, including dialing the number and putting the number into an electronic telephone book. Upon selection of the action for putting the number in an electronic telephone book, user interface **240** transmits the corresponding telephone number and selected action to action processor **250**. Action processor **250** locates and opens the electronic telephone book, places the telephone number in the appropriate field and allows the user to input any additional information into the file.

FIGS. **8** and **9** display a flowchart illustrating preferred method **800** for recognizing patterns in documents and performing actions. This method is carried out during the run-time of application **167**. Referring first to FIG. **8**, method **800** starts by receiving **810** the content, or a portion of the content, from document **210**. Assuming program **165** initiates with the receipt of any text, the received content or portion is scanned **820** for identifiable structures using the patterns in analyzer server **220**. Upon detection of a structure based on a particular pattern, actions associated with the particular pattern are linked **825** to the detected structure. Assuming a display-type environment, the presentation region location for a detected structure is retrieved **830** from application **167**. If the document content being displayed on output device **105** is changed **840**, for example by the user adding or modifying text, method **800** restarts. Otherwise, method **800** continues with block **850**. If the presentation regions change **850**, for example by the a user scrolling document **210**, then new presentation regions from application **167** are again retrieved **830**. Otherwise, method **800** continues to block **860**. As illustrated by block **860**, method **800** loops between blocks **840** and **860** until a request for display of identified structures is received **860**. It will be appreciated that the steps of the loop (blocks **840**, **850** and **860**) can be performed by application **167**.

Referring also to FIG. **9**, when a request for the display of detected structures is received **860**, the regions are displayed **910** using presentation mechanisms such as highlighting the presentation region around each detected structure, although alternative presentation mechanisms can be used. If a request for the display of candidate actions linked to a detected structure is not received **920**, method **800** returns to block **840**. However, if a request is received **920**, the actions linked in block **825** are displayed **930**. This request for display of candidate actions can be performed using a selection mechanism, such as a mouse-down operation over a detected structure, which causes the candidate actions linked to the structure to be displayed **930**. Display **930** of candidate actions may be implemented using a pop-up menu, although alternative presentation mechanisms can be used such as pull-down menus, dialog boxes and voice synthesizers.

As illustrated in block **940**, if an action from the displayed candidate actions is not selected **940**, method **800** returns to block **840**. However, if an action is selected **940**, the action is executed **950** on the structure selected in block **920**. After execution **950** of an action, method **800** returns to block **840**. Method **800** ends when the user exits application **167**, although other steps for ending method **800** can alternatively be used.

Referring now to FIG. **10**, a flowchart illustrating the preferred method **820** for scanning and detecting patterns in a document is shown. Method **820** starts by retrieving **1010** data to be analyzed. After the data is retrieved, several pattern analysis processes may be performed on the data. As illustrated in block **1020**, a parsing process retrieves **1030** grammars, detects **1040** structures in the data based on the retrieved grammars, and links **1050** actions associated with each grammar to each structure detected by that grammar. As illustrated in block **1060**, a fast string search function retrieves **1070** the contents of string library **420**, detects **1080** the strings in the data identical to those in the string library **420**, and links **1090** actions associated with the library string to the detected string. As illustrated in block **1100**, additional pattern analysis processes, such as a neural net scan, can be performed **1100** to detect in the data other patterns, such as pictures, graphs, sound, etc. Method **820** then ends. Alternatively, the pattern analysis processes can be performed in parallel using a multiprocessor multitasking system, or using a uniprocessor multithreaded multitasking system where a thread is allocated to execute each pattern detection scheme.

These and other variations of the preferred and alternate embodiments and methods are provided by the present invention. For example, program **165** in FIG. **1** can be stored in ROM, disk, or in dedicated hardware. In fact, it may be realized as a separate electronic circuit. Other components of this invention may be implemented using a programmed general purpose digital computer, using application specific integrated circuits, or using a network of interconnected conventional components and circuits. The analyzer server **220** of FIG. **2** may use a neural net for searching a graphical document **210** for faces, or a musical library for searching a stored musical piece **210** for sounds. The user interface **240**

may present structures and actions via voice synthesis over a telephone line connection to system **100**. The embodiments described have been presented for purposes of illustration and are not intended to be exhaustive or limiting, and many variations and modifications are possible in light of the foregoing teaching. The system is limited only by the following claims.

What is claimed is:

1. A computer-based system for detecting structures in data and performing actions on detected structures, comprising:

an input device for receiving data;

an output device for presenting the data;

a memory storing information including program routines including

an analyzer server for detecting structures in the data, and for linking actions to the detected structures;

a user interface enabling the selection of a detected structure and a linked action; and

an action processor for performing the selected action linked to the selected structure; and

a processing unit coupled to the input device, the output device, and the memory for controlling the execution of the program routines.

2. The system recited in claim **1**, wherein the analyzer server stores detected structures in the memory.

3. The system recited in claim **1**, wherein the input device receives the data from an application running concurrently, and wherein the program routines stored in memory further comprise an application program interface for communicating with the application.

4. The system recited in claim **1**, wherein the analyzer server includes grammars and a parser for detecting structures in the data.

5. The system recited in claim **4**, wherein the analyzer server includes actions associated with each of the grammars, and wherein the analyzer server links to a detected structure the actions associated with the grammar which detects that structure.

6. The system recited in claim **1**, wherein the analyzer server includes a string library and a fast string search function for detecting string structures in the data.

7. The system recited in claim **6**, wherein the analyzer server includes actions associated with each of the strings, and wherein the analyzer server links to a detected structure the actions associated with the grammar which detects that string structure.

8. The system recited in claim **1**, wherein the user interface highlights detected structures.

9. The system recited in claim **1**, wherein the user interface enables selection of an action by causing the output device to display a pop-up menu of the linked actions.

10. The system recited in claim **1**, wherein the programs stored in the memory further comprise an application running concurrently that causes the output device to present the data received by the input device, and an application program interface that provides interrupts and communicates with the application.

11. The system recited in claim **1**, wherein the user interface enables the selection of a detected structure and a linked action using sound activation.

12. The system recited in claim **1**, wherein a first one of the actions may invoke a second one of the actions.

13. A program storage medium storing a computer program for causing a computer to perform the steps of:

receiving computer data;

detecting a structure in the data;

linking at least one action to the detected structure;

enabling selection of the structure and a linked action; and

executing the selected action linked to the selected structure.

14. In a computer having a memory storing actions, a system for causing the computer to perform an action on a structure identified in computer data, comprising:

means for receiving computer data;

means for detecting a structure in the data;

means for linking at least one action to the detected structure;

means for selecting the structure and a linked action; and

means for executing the selected action linked to the selected structure.

15. In a computer having a memory storing actions, a method for causing the computer to perform an action on a structure identified in computer data, comprising the steps of:

receiving computer data;

detecting a structure in the data;

linking at least one action to the detected structure;

enabling selection of the structure and a linked action; and

executing the selected action linked to the selected structure.

16. The method recited in claim **15**, wherein the computer data is received from the application running concurrently.

17. The method recited in claim **15**, wherein the memory contains grammars, and wherein the step of detecting a structure further comprises the steps of retrieving a grammar and parsing the data based on the grammar.

18. The method recited in claim **17**, wherein the grammar is associated with a particular action, and wherein the step of linking at least one action to the detected structure includes the step of linking the particular action to the detected structure.

19. The method recited in claim **15**, wherein the memory contains strings, and wherein the step of detecting a structure further comprises the steps of retrieving a string from the memory and scanning the data to identify the string.

20. The method recited in claim **15**, further comprising after the step of detecting a structure, the step of highlighting the detected structure.

21. The method recited in claim **15**, further comprising, after the step of linking at least one action to the detected structure, the step of displaying and enabling selection of an action for performance on the detected structure.

22. A computer-based method for causing a computer to identify, select and perform an action on a structure in computer data received from a concurrently running application, said application presenting the computer data to the user, the method comprising the steps of:

receiving computer data from the application;

detecting a structure in the computer data;

linking at least one action to the detected structure;

communicating with the application to determine the location of the detected structure as presented by the application, to enable selection of the detected structure and a linked action, and to determine if the detected structure and a linked action have been selected; and

performing a selected action linked to the detected pattern.

23. The method recited in claim **15**, wherein the step of enabling uses sound activation.

24. The method recited in claim **15**, wherein a first one of the actions may invoke a second one of the actions.

* * * * *

CERTIFICATE OF SERVICE

I hereby certify that I electronically filed the foregoing with the Clerk of the Court for the United States Court of Appeals for the Federal Circuit by using the appellate CM/ECF system on January 31, 2014.

I certify that all participants in the case are registered CM/ECF users and that service will be accomplished by the appellate CM/ECF system.

Dated: January 31, 2014 /s/ Deanne E. Maynard

CERTIFICATE OF COMPLIANCE WITH RULE 32(a)

This brief complies with the type-volume limitation of Rule 32(a) of the Federal Rules of Appellate Procedure because it contains 12,347 words.

Dated: January 31, 2014 /s/ Deanne E. Maynard